CROSSTALK

WORKING AS A TEAM

Report Documentation Page

1. REPORT DATE **NOV 2007**	2. REPORT TYPE	3. DATES COVERED **00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE **CrossTalk: The Journal of Defense Software Engineering. Volume 20, Number 11, November 2007**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **OO-ALC/MASE,6022 Fir Ave,Hill AFB,UT,84056-5820**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**32**	

Working as a Team

Software Engineering Technology

ON THE COVER

Cover Design by
Kent Bingham

Additional art services
provided by Janna Jensen

Departments

Working as a Team

B ack in May, I was watching the Giro d'Italia and the incredible selfless sacrifice of American cyclist David Zabriskie. Zabriskie was arguably one of the strongest riders in the peloton, but his teammate Andy Schleck was even stronger and stood the best chance of winning the race. Despite Zabriskie's chance to place high in the overall standings, he rode to help his teammate conserve energy and finish the race second on the podium. He and every other member of the team were focused on the performance of the team, not the performance of the individual. Effective software development teams need to have the same mutual commitment to work together towards a common goal. This level of mutual commitment doesn't happen just by telling people that we want them to work as a team. Such commitment requires several enablers. Many of the articles in this month's addition will touch upon processes for creating teams that truly work together. I'd like to offer my brief thoughts on four key enablers for creating mutual commitment of team members.

1. **Measure the project.** Over and over again I see proof that *what gets measured gets done*. If you focus your measurements on individuals, then individuals will perform. If you focus your measurements on project performance, the project will perform. Measures should focus on what is important to the customer and customers don't care about the performance of individuals. We all have internal measures and targets, but leaders should insulate their teams so they can focus on those that matter to their customer.

2. **Reward project performance not the individual.** Although easier said than done, especially in the public sector, it is the performance of the project that should be incentified not the performance of the individual. If reward incentives focus on the individual, there is little incentive to make someone else look good. Project incentives will help individuals focus on the critical path, not just their piece of the project.

3. **Limit specialization.** Rather than have developers narrowly focused on one specialty which supports many projects, utilize people broadly to avoid bottle necks and encourage ownership. One person can not have the same level of commitment to many projects as the team members who are dedicated to one project. When possible it is better to have them committed to one or two teams and utilized in areas outside of their expertise to supplement the project.

4. **Institute Team Member Radio.** In my earlier cycling analogy, each team member is equipped with a radio that is in contact with the team director (and other coaches) in the team car. When a key member of the team suffers from a flat or has a mechanical problem, the team director signals the team to drop back to help their team member rejoin the peloton. The director also keeps close watch on the details of the race. The director warns the team of obstacles and watches for threatening tactics from the other teams. The team members themselves are communicating to let the director know what is going on from inside the peloton that can't be seen from the team car. When a team member might be suffering more than they care to admit, his team members can communicate information back to the director so she can decide what to do. This is exactly how a cohesive software development team should work. When a key area or key member of a software team gets in trouble the team needs to come to the rescue. The only way for the team to avoid obstacles, capitalize on opportunities, and react to trouble is to have constant communication up, down, and through the team.

Speaking of teamwork, we have a great team of authors lined up for this month. Starting with LTC Nanette Patton and Allan Shechet, they address achieving project success in *Wisdom for Building the Project Manager/Project Sponsor Relationship: Partnership for Project Success*. Jennifer Tucker and Hile Rutledge describe the necessity of maximizing team performance in *Shaping Motivation and Emotion in Technology Teams*, and Kasey Thompson and Tim Border teach managers to build more efficient teams in their article *The Gauge That Pays: Project Navigation and Team Building*.

Kevin Stamey
Oklahoma City Air Logistics Center, Co-Sponsor

Wisdom for Building the Project Manager/Project Sponsor Relationship: Partnership for Project Success

LTC Nanette Patton
Office of the Surgeon General

Allan Shechet
Savvy Services Incorporated

The project sponsor can promote information technology (IT) project success in several ways, yet many projects either have no formally designated project sponsor or the project sponsor is confused about his/her role. The project sponsor's role traditionally includes project approval, funding, and staffing, but can include much more. The project sponsor is sometimes called the champion of the project or the key stakeholder of the project. Because the role of the project sponsor sometimes overlaps with the project manager's role, confusion can arise. This article discusses conventional roles and responsibilities of the project sponsor and then discusses strategies a project manager can employ to define boundaries to reduce role confusion and promote partnership to facilitate project success.

What is the role of the project sponsor? Projects fail for a plethora of reasons. One of the most common reasons for project failure is lack of project management discipline. Another which is much harder to rectify is cultural resistance to change [1]. These reasons identify the point of demarcation between the project manager and the project sponsor. The project manager runs the project on a day-to-day basis to produce a solution to a business problem, and the project sponsor manages the organizational culture to ensure it is ready to receive, accept, and implement that solution. If success is about getting results, then the role of the project sponsor – in the simplest terms – is to help the project manager ensure the project achieves success through desired results.

Thomsett International has reviewed more than 20 major projects that were in the process of failing or had failed. They did these reviews not as an academic exercise or a controlled experiment, but rather these reviews were undertaken *in the heat of the battle*. In every one of the 20 major failed projects, lack of an effective sponsor was a common deficiency. As one project manager recently put it: *To manage a project without an effective executive sponsor is to visit hell on Earth* [2].

If you have ever been a project manager, then perhaps you have come to realize the limitations of your influence because no matter how well trained or experienced you may be, corporate objectives associated with your project will never be achieved if there is not someone leading and directing the business change. Back in 1515 when he published "The Prince," Nicolò Machiavelli said the following:

And it ought to be remembered that there is nothing more difficult to take in hand, more perilous to conduct, or more uncertain in its success, than to take the lead in the introduction of a new order of things. Because the innovator has for enemies all those who have done well under the old conditions, and lukewarm defenders in those who may do well under the new. This coolness arises partly from fear of the opponents, who have the laws on their side, and partly from the incredulity of men, who do not readily believe in new things until they have had a long experience of them. Thus it happens that whenever those who are hostile have the opportunity to attack, they do it like partisans, whilst the others defend lukewarmly, in such wise that the prince is endangered along with them. [3]

The perils associated with creating *a new order of things* are just as true today as they were back in Machiavelli's day. And this is where the project sponsor comes in to assist the project manager in the shared responsibility of delivering both project deliverables and project outcomes.

So what does managing the culture entail? Neil Love and Joan Brant-Love define the project sponsor's role as mentor, catalyst, cheerleader, barrier buster, boundary manager, and senior management liaison [4].

1. **Mentor**
   - Increases the confidence of the project manager.
   - Helps the project manager understand the full business context of project decisions.
   - Improves the project manager's leadership and problem-solving skills.
2. **Catalyst**
   - Stimulates the thinking and perspectives of the project manager.
   - Challenges assumptions.
   - Plays *devil's advocate* to help the project manager see more options/reactions and raises the level of thinking of the project manager.
3. **Cheerleader**
   - Helps the project manager and others stay motivated and deal with team issues.
   - Occasionally directly helps the team members stay motivated through pep talks and celebrations.
   - Reminds the project manager and the team of the importance of the mission.
4. **Barrier Buster**
   - Knocks down barriers that are beyond the control of the project manager or project team.
   - Barriers can include non-supportive senior managers and managers of team members, resource problems, team member availability problems, or lack of tools/equipment/facilities/software needed by the team.
5. **Boundary Manager**
   - Keeps executives, managers, and professionals from meddling or interfering with the team's progress.
   - Protects the team from unnecessary interactions with others or unnecessary reporting to others.
   - Lets the team perform within the boundaries of the agreed-to team mission and contract.
6. **Senior Management Liaison**
   - Before establishing a team, the project sponsor briefs the organi-
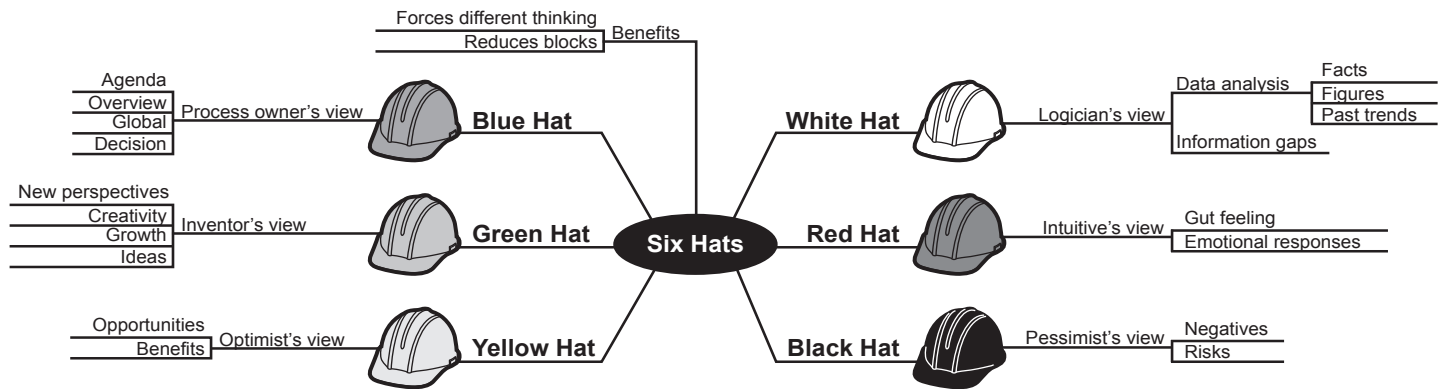
The de Bono Group's *Six Thinking Hats®* is property of The de Bono Group.

Figure 1: *The de Bono Group's Six Thinking Hats Topics* [9]

zation's senior leadership group on the planned team's mission, desired team members, and any constraints on the project.

- As the team moves through the project life cycle, the project sponsor periodically communicates to management the team's progress to date, and asks for help on issues beyond the control or influence of the project sponsor.

Breaking it down further, responsibilities of the project sponsor can include the following:

- Ensuring the project manager and the team are aligned to a common purpose.
- Providing guidance for key business strategies.
- Empowering the project manager.
- Championing the project and the team.
- Formally managing the project's scope.
- Approving plans, schedules, and budgets.
- Ensuring sustained buy-in.
- Clearing roadblocks.
- Ensuring timely availability of resources.
- Reviewing the project's progress.
- Ensuring that project benefits are realized.
- Leading the functionals through business process re-engineering.

So, another way of looking at the division of role and responsibilities between the project manager and the project sponsor is the project sponsor plays a more strategic role helping the team and the rest of the organization understand how the project supports the strategic plan of the organization. The project manager plays a more tactical role and is responsible for the day-to-day progress of the project.

What Can the Project Manager Do to Make Sponsorship Work?

Technology pull or technology push cre-

ates different conditions for project initiation and consequently the staffing of the project. In the technology pull scenario, it is more likely the project sponsor will manifest prior to the project manager. As the project and project goals evolve, the project manager will have to assess whether or not that initial person driving the adoption of the technology remains the best candidate for continued project success. In the technology push scenario, the project manager will more likely appear before the sponsor. This scenario affords the opportunity for the project manager to work with the organization's leadership to help identify and select the sponsor.

The person identified as the sponsor must be high enough in the organizational hierarchy to have organizational authority commensurate with the scope of the project. Conversely, the project sponsor cannot be too high in the organization such that he or she does not have sufficient time to dedicate to the project or is too far removed from the scope and objectives of the project to be effective in giving direction. The logical project sponsor is the person in the organization who both wants the project accomplished and has responsibility for all of the organizational units affected. In short, the sponsor is the person who can make it happen.

Further complicating matters is the possibility that the project sponsor and manager may come from different organizations. For example, project managers for United States Army Medical Command (USAMEDCOM) enterprise-wide projects are assigned from the U.S. Army Medical Information Technology Center (USAMITC) and the project sponsor (functional proponent) is typically assigned from another command within the USAMEDCOM where the functional expertise resides. This dynamic

can lead to conflict from differing cultures and loyalties that will have to be reconciled.

Once the project sponsor is identified and teamed with a project manager, the project manager needs to commit to taking responsibility for the quality and productivity of the manager/sponsor relationship to meet project accountabilities [5]. Developing a good manager/sponsor team is not an accident; rather, it is a function of the manager's relationship behavior (the manner in which the manager relates to the sponsor, which in turn creates a response that sets the tone of the relationship). The project manager must take personal responsibility for the quality of that relationship, never waiting for senior leadership to notice and act on a situation that needs attention. To do that, the project manager must do the following:

- Recognize individuals are not passive recipients in the team/partnership experience, that individual behavior shapes every team, and that the individual affects the team at least as much as the team affects the individual.
- Acknowledge that not attending to team performance and the project manager/sponsor relationship is a choice and puts the project manager at the mercy of chance.
- Accept in the scenario of project sponsor/manager shared responsibility that the quality and productivity of that relationship is worthy of focus.
- Learn what behaviors and processes lead to a successful partnership and exhibit them [5].

Conversely, the project sponsor must take an active role and may be required to back that role up with travel, late nights, and presentations to stakeholders. First time sponsors typi-

cally underestimate the amount of time and commitment it takes, and the seasoned project manager will be prepared with estimates and confirmation from other project sponsors. Project managers, more typically devoted to a single project, must take into consideration that the sponsor may be covering multiple projects. However, there is an upper limit for the project sponsor. If the sponsor covers too many projects at once, effectiveness can become diluted.

Dr. Ted Weston, Jr., Professor in the Computer Information Systems Department of Colorado State University's College of Business, cannot overemphasize the importance of a single project sponsor:

> Without an empowered sponsor, the project is essentially DOA [dead on arrival]. In order to get the project support needed, the sponsor must be a part of top management and have the ear of top management. The vision of the sponsor (and there must be one) and the vision/expectations of the stakeholders must be essentially one and the same. The project steering committee, by whatever name, must be supportive of both the sponsor and the project's goals. If wishy-washy – STOP. If functional area or sponsor is not respected, the project is off to a bad start. Furthermore, the sponsor's commitment to the project must be real and not lip-service. If you cannot sense the *fire in the belly* of the sponsor – stop! The sponsor must actively champion the project and communicate the sense of project urgency. No urgency – the project is again in trouble. [6]

Both the project manager and sponsor must recognize what effective sponsorship looks like. Project sponsorship is far more than saying, *here's a bunch of resources, tell me when we're better*. Many project sponsors have little understanding of the role, and it may fall to the project manager to educate and coach them – on both the tasks and the time commitment.

Some organizations offer some kind of training for the project sponsor. For example, the Army Medical Department (AMEDD) offers the Health Systems Functional Proponent Course (HSFPC) to help prepare project sponsors for the demands of the role [7]. Dr. Barbara Erickson, one of the cre-

ators of the HSFPC, said the following:

> The HSFPC was established in response to the education and training assessment resulting from the Task Force Mercury Reengineering Study of 1996, which was commissioned by The Army Surgeon General (then Major General James Peake). The task force identified Information Management (IM)/Information Technology (IT) requirements and gaps. Subsequently, Information Manage-

> *"Both the project manager and sponsor must recognize what effective sponsorship looks like. Project sponsorship is far more than saying, 'here's a bunch of resources, tell me when we're better.' ... it may fall to the project manager to educate and coach them ..."*

> ment/Information Technology Subject Matter Experts and training experts worked together to identify the critical tasks, skills, and knowledge needed to meet deficiencies that could be corrected by education and training and thus the HSFPC was born to serve as the missing link between the functional (or business process) requirements and the technical realization of them. [7]

USAMITC is the AMEDD's execution, acquisition, and materiel development arm for information technology. In this role, USAMITC houses the project managers for the AMEDD enterprise IT project who partner with the functional proponents (aka project sponsors) for the delivery of a new IT

system or service. When LTC Patton was the Chief Information Officer (CIO) at USAMITC, she made a point of personally presenting at the HSFPC. Prior to her arrival, the block of instruction on USAMITC and its services was taught by a USAMITC marketing specialist. Mission success for USAMITC is in part derived from successful partnerships between the USAMITC project directors and the functional proponents. Knowing the graduates of the course would be paired with project managers from USAMITC, she decided advocating and facilitating this relationship was too important to delegate to a marketing specialist. Thus, project managers should enlist their organization's IT leadership in promoting successful partnerships between project sponsors and managers.

If the organization is less formal in training project sponsors, the project manager can refer the sponsor to other reading material. For example, the California Office of Systems Integration posts its best practices on its public Web site, which includes a listing of the roles and responsibilities of the project sponsor in their work environment [8].

Negotiating the Sponsor Agreement

While the roles and responsibilities cited earlier are commonly associated with the project sponsor, each project is different and there is always room for negotiation. Depending on the situation at hand and the knowledge, skills, and abilities of the people involved, roles can be assigned differently (see Figure 1, page 5). Each person is different. Each person has a unique set of talents, a unique pattern of behaviors, passions, and yearnings. Each person's pattern of talents is enduring and resistant to change. Each person has a unique destiny. *The goal is to help each person become more of who he already is* to maximize benefit for the manager/sponsor partnership and the project team [9]. Thus, the project manager can and should work with the sponsor to develop a sponsorship agreement, clarifying the sponsor's role and specific responsibilities and identifying role boundaries between the sponsor and the project manager, thereby reducing the confusion created by the occasional overlap of the roles and leveraging individual strengths to promote project success.

Even the most educated and experienced managers occasionally argue or misunderstand each other; however, the

probability of this occurring can be minimized by laying out in writing the roles and responsibilities of the project sponsor. Documenting delineated roles and responsibilities is particularly important when sponsorship has been delegated to an acting sponsor – some responsibilities may be delegated and others may remain with the logical (original) sponsor.

The entire project team must agree on both the unique and shared responsibilities of the project sponsor and project manager based on what the organization and project team agree is important and feasible.

Table 1 represents a standard template the team can use to start the negotiation process [10, 11].

As the project progresses, the project manager can take specific steps to keep the sponsor involved throughout the life of the project. Again, knowing what the sponsor should be doing and clearly stating those expectations is primary. The project manager can facilitate sponsor involvement in the following ways:

- As the project moves from one phase to another, discuss with the sponsor what to expect and what questions he/she should be concerned with.
- When deliverables are submitted for review, indicate what kind of feedback is needed.
- Involve the sponsor in the preparations for major project reviews, emphasizing decisions required for progress.
- Inform the sponsor promptly of issues needing sponsorship resolution, providing background, pros/cons, and recommendations (see Table 2, next page).

Real-Life Applications

Approaching the project manager/sponsor relationship as a true partnership, the project manager and project sponsor can divide specific roles given the general guidelines provided according to each partner's strengths and non-strengths. For example, back in 2000, USAMITC was embarking on its implementation of PlanView, an enterprise project management and human resource planning tool. Operation of the tool requires employees to submit timesheets in which they record their time against tasks associated with project work breakdown schedules. Initially the assigned project manager worked alone without the help of a project

Title	Role
Project Manager	The person responsible for developing, in conjunction with the project sponsor, a definition of the project. The project manager then ensures that the project is delivered on time, to budget, and to the required quality standard (within agreed specifications). He/she ensures the proper allocation of resources within the project, and manages relationships with a wide range of groups (including all project contributors). The project manager is also responsible for managing the work of consultants, allocating and utilizing resources in an efficient manner, and maintaining a cooperative, motivated, and successful team.

Responsibilities
- Managing and leading the project team.
- Recruiting project staff and consultants.
- Managing coordination of the partners and working groups engaged in project work.
- Using detailed project planning and control, including the following:
  - Developing and maintaining a detailed project plan.
  - Managing project deliverables in line with the project plan.
  - Recording and managing project issues and escalating where necessary.
  - Resolving cross-functional issues at project level.
  - Managing project scope and change control and escalating issues where necessary.
  - Monitoring project progress and performance.
  - Providing status reports to the project sponsor.
  - Managing project training within the defined budget.
  - Providing liaison with, and updates progress to, project steering board/senior management.
  - Managing project evaluation and dissemination activities.
  - Managing consultancy input within the defined budget.
  - Providing final approval of the design specification.
- Working closely with users to ensure the project meets business needs.
- Providing a definition and management of the user acceptance testing program.
- Identifying user training needs and devising and managing user training programs.
- Accepting ultimate authority and responsibility for project outputs/deliverables.

Title	Role
Project Sponsor	The person who commissions others to deliver the project and champions the cause throughout the project. Sponsors will normally be a senior member of staff with a relevant area of responsibility that will be affected by the outcome of the project. They are involved from the start of the project, including defining the project in conjunction with the project manager. Once the project has been launched, they should ensure that it is actively reviewed. The project sponsor is usually the one who has to negotiate a path through the tricky diplomatic areas of the project.

Responsibilities
- Acts as champion of the project, ensuring it is adequately resourced for success.
- Ensures the timely availability of these essential project resources.
- Develops and communicates a vision and direction.
- Sponsors the communications program; communicates the program's goals and relates them to the strategic goals and objectives of the organization as a whole.
- Behaves congruently and provides a consistent message.
- Promotes participation and inclusiveness particularly within the business unit in which the solution will be implemented and assists the project manager in promoting this within the project team.
- Gets things done through commitment rather than control.
- Ensures resolution of issues escalated by the project manager or the project board.
- Makes key organization/commercial/business decisions for the project.
- Approves the budget and decides tolerances; recommends killing the project if appropriate.
- Leads the project steering board.
- Accepts authority and responsibility for achieving the desired business outcomes derived from the project.
- Is accountable for the timely delivery of planned benefits associated with the project.

Table 1: *Negotiation Process Standard Template*

sponsor to facilitate implementation. However, that arrangement did not produce the desired results as the implementation was not adequately progressing for the satisfaction of the commander. The commander decided to assign one of the division chiefs as

the new project manager.

While this division chief had the title of *project leader*, in reality she assumed the role of project sponsor and she partnered with the previous project leader, delegating the traditional day-to-day roles of the project manager

Project Phase	Concerns
Initiation	• Why is this project needed? What's the business problem being solved or the opportunity to be seized? How does the project support our corporate goals? • Do we have a long-term vision and investment strategy in place? • From the organizational perspective: What are the objectives? What will the end result look like? • From the employee perspective: What are the benefits? How will life be better when the project is over? • How will we measure success? What is our baseline? What is our target? • What areas of the organization will be affected? In what ways? • What are the constraints – in time, in money, in quality? • What can realistically be achieved within those constraints? • Should we proceed?
Planning	• Who needs to be involved? And how? Are they adequately trained? • What are the boundaries or scope of the project? • Roughly how much will it cost and how long will it take? • What are the risks? Can they be managed? • Do we have metrics and processes in place to promote project success? Do we have systems in place to generate quantifiable data and demonstrable knowledge to create go/no-go decisions? • Is the team incentivized on forward-looking measures? • Have we assembled the right team? Are team members aligned to a shared purpose? • Should we proceed?
Execution	• Are we accomplishing what we planned to accomplish? Within the planned time frame? With the planned resources? Within budget? • Is there anything I can do to facilitate the team's work? • Are we getting the cooperation we need from the business units? • What can I do to remove obstacles and promote the provision of the right level of support so the project manager has a clear path for project execution? • Does senior leadership understand their role and responsibilities in IT project management process? What can I do to facilitate understanding?
Control	• Are plans in place to measure the predicted benefits? • Have we integrated cost, project control, and knowledge management?
Closing	• Did we accomplish what we planned to accomplish? Within the planned time frame? With the planned resources? Within budget? • How did we perform based on our success criteria? • What lessons did we learn? • What remains to be done?

Table 2: *Sponsor Concerns by Project Phase* [12, 13]

to the old project leader who remained part of the project team. The previous leader became the *technical lead* and she became the *functional lead* handling the business process reengineering, training, and project promotion. This arrangement leveraged each other's experience, power position, and strengths.

With a partnership established and the roles clearly delineated between the technical lead and the functional lead, she set about to expedite implementation and user adoption of the system, focusing on the desired business outcomes of the commander as a project sponsor is expected to do. To those ends, she developed a metrics program to direct time and energy to activities that would hasten project success. Meanwhile, the technical lead focused on implementing software updates and upgrades, pushing the software to the user base and server management. Initially, each week in the commander's staff meeting with the division chiefs, the functional lead would report by division the percentage of staff members completing their timesheets by the designated deadline. If the division did not meet a certain percentage, they were labeled *red*. She continued to focus on this metric until it was mostly *green* across the board and then she changed the key metric.

As a project management organization, the USAMITC leadership had to ensure people were devoting most of their time to project work rather than overhead functions in order to demonstrate value to the AMEDD enterprise. While she continued to report on timesheet reporting compliance as a control measure, she now diverted the attention of the division chiefs to where their staff was reporting their time. Again, thresholds were set and divisions were given a red, amber, or green rating. As a result of this focused attention, the project team discovered additional categories needed to be added to the system for reporting project work. As these shortcomings and obstacles to adoption were uncovered by the functional lead through her analysis of the metrics and follow up on them with the other division chiefs, the technical lead then modified the customization design for the tool based on these lessons learned. The functional lead continued this cycle of devising and introducing new metrics designed to streamline the business operation of the system until she had to move onto her next assignment. Her successor as project leader continued the same division of responsibilities and expanded her metrics program.

When Raytheon purchased 80 percent of Hughes Aircraft Company, a large project was created for Computer Sciences Corporation to separate the Raytheon portion of the information technology infrastructure from the Hughes Aircraft Company infrastructure. A senior project manager who worked for a program manager responsible for a portfolio of projects saw a dramatic demonstration of the power of a sponsor when he was asked by the program manager to improve the disastrous results realized from a pilot project for desktop changes. The schedule was very tight and the program manager wanted a revised process rolled out the following week. The senior project manager went to the project sponsor that was the process owner and explained the problem. They agreed additional staff (two) and time (six weeks) was needed and the sponsor called the program manager and negotiated additional budget and schedule time to revise the processes and do another pilot. The revised process was a tremendous success, and the time spent creating the process was more than made up in the ensuing months of changes contributing to a program completed ahead of schedule and under budget with minimal disruption to the users. The sponsor's willingness to play an active role in the project, along with his credibility and position was essential in negotiating the additional staff and time to do the job right.

Conclusion

In the shared responsibility of project success, the project manager focuses on deliverables and the project sponsor focuses on outcomes. Keeping this division of labor

clear, the project sponsor can make the project manager's life immeasurably easier and greatly improve the odds for project success. It is a partnership worthy of the project manager's time, focus, and effort.◆

References

1. Lundberg, Abbie. "The Hard Road To Change." CIO, 2005 <www.cio.com/archive/061505/edit.html>.
2. Thomsett International. "Causes, Patterns and Symptoms of Project Failure." 2005 <www.thomsettinternational.com/main/articles/path/pathology.htm>.
3. Machiavelli, Nicolò. The Prince. 1505.
4. Love, N., and J. Brant-Love. "The Project Sponsor Guide." Newtown Square: Project Management Institute (PMI), 2002.
5. Avery, Christopher M., Meri Aaron Walker, and Erin O'Toole. Teamwork Is an Individual Skill: Getting Your Work Done When Sharing Responsibility. Berrett-Koehler Publishers, 2001.
6. Weston, Ted. E-mail to LTC Patton.
7. United States. AMEDD Center and School Portal 8 Aug. 2007 <www.cs.amedd.army.mil/details.aspx?dt=149>.
8. State of California. Office of Systems Integrations. 8 Aug. 2007 <www.bestpractices.cahwnet.gov/ProjectOfficeRoles.aspx?rid=2&pid=0>.
9. Buckingham, Marcus, and Curt Coffman. "First, Break All the Rules: What the World's Greatest Managers Do Differently." What Can You Do With a Major in Business: Real People. Real Jobs. Real Rewards. Simon and Schuster, 1999.
10. JISC infoNet. "Project Roles and Responsibilities." <www.jiscinfonet.ac.uk/InfoKits/infokit-related-files/roles-and-responsibilities-template>.
11. Avery, Christopher M. "The Leader's Guide." 2001 <www.christopheravery.com/store/free_leaders_guide.htm>.
12. United States. Government Accounting Office (GAO). "Better Support of Weapon System Program Managers Needed to Improve Outcomes." GAO-06-110. Washington: GAO, 2005 <www.gao.gov/new.items/d06110.pdf>.
13. Knutson, Joan. "The Role of the Project Sponsor: Committed, Clear and Connected." Chief Project Officer <www.chiefprojectofficer.com/article/55>.

About the Authors

LTC Nanette Patton is currently serving as a program analyst in the Office of the Surgeon General and is detailed to the Department of Defense (DoD) Task Force on the Future of Military Health Care. She has over eight years experience as a health services systems manager and is Level III DoD-Acquisition certified in IT and Level II certified in program management. Patton has a master's degree in human resources management from Chapman University and a master's degree in business administration with a concentration in computer information systems from Colorado State University.

DoD Task Force on the Future of Military Health Care
5205 Leesburg Pike
Skyline 1, STE 1000A
Falls Church, VA 22041
Phone: (703) 681-8910 ext. 1003
E-mail: nanette.patton@us.army.mil

Allan Shechet is president of Savvy Services Incorporated, a project management consulting company, and he has more than 20 years aerospace and IT project management experience. Shechet has a master's degree in organization development from Fielding Graduate University, is certified as a Project Management Professional by the PMI, and is an active volunteer for their Los Angeles chapter.

Savvy Services Incorporated
2239 Linnington AVE
Los Angeles, CA 90064
Phone: (310) 720-2522
E-mail: allan@savvyservices.net

Shaping Motivation and Emotion in Technology Teams

Jennifer Tucker and Hile Rutledge
Otto Kroeger Associates

In previous articles for CROSSTALK[1]*, we described the personality dynamics of Information Technology teams and presented a diagnostic model for analyzing the human dynamics of large systems development programs. In this article, we discuss the role of motivation and emotion in maximizing team performance and present an actionable and accessible approach for shaping both motivation and emotion – in self, in others, and in teams.*

The ability to motivate others is a critical skill for anyone leading a technology team, and motivation lies at the heart of any action that someone takes. When we talk about influencing a technology partner, inspiring a team to take a risk, or managing people toward *peak performance*, we are really talking about motivating another person towards some desired end. Motivating others is an act of leadership.

Emotions are a natural byproduct of motivation. When we are motivated to act in a certain way and those motives are satisfied, we feel good emotions such as pride, belonging, gratitude, relaxation, and excitement. When we are motivated to act in a certain way and those motives are not fulfilled, we feel negative emotions such as anxiety, resentment, guilt, and shame.

Emotions are a natural part of being human and they help define who we are. Despite this, we often see technologists try to keep these emotions *under wraps* – as evidenced by the common phrases *let's not get emotional about this* and *let's stay objective, people*.

We believe that by better understanding and actively shaping motivation and emotion, we can lead teams to greater success. This article provides a practical and proven methodology for first recognizing the motivational states and resulting emotions that both help and hinder team effectiveness and then altering those states in order to produce different emotions.

Linking Motivation and Emotion to Performance

For many, motivation and emotion are seen as somewhat messy and hard to control, which can lead to de-emphasizing their importance in the quest for a more objective and impersonal approach to technology problems. In the end, when everything must be converted to the code of a technology world, the ambiguity of emotions seems like something to avoid.

Consider, however, how some of the specific challenges we have seen in our work with technology teams link to motivation and emotion:

- The call to meet impossible deadlines amidst scope creep and oversight scrutiny leads to high levels of anxiety and turnover on a mission-critical software development team.
- A team that fails to adequately integrate stakeholders early and frequently enough into the system-development process admits a generally low level of empathy for their users.
- A program team that has failed to plan effectively for integration with another system in a *system of systems* effort professes strong feelings of competition with the *sister* program.
- Developers on a team feel so pressured to meet their sponsor's marketing spin that they fail to speak up when security requirements are compromised in the name of *usability*.

We have all seen how personal motives can impact a work product. Recognizing the link between motivation and performance, however, is only a first step. We need a framework to help systematically detect and then alter these motives, leading to more productive outcomes.

Understanding Motivational States

One valuable framework is Reversal Theory (Table 1), a powerful set of ideas that casts a unique light on human motivation, emotion, and behavior. Reversal Theory is a psychological theory addressing the flexibility and changeability of individuals. The theory specifically focuses on motivation, proposing that people regularly reverse between opposing psychological states, depending upon the meaning and motives felt in different situations at different times. These reversals are healthy and necessary – as situations and meanings change, so do motives and emotions [1].

Reversal Theory proposes that key emotions (such as anger and fear) and values (such as achievement and control) can be traced to different motivational states, which operate in pairs along four different focus areas called *domains*. We spend our lives moving between the different motivational states in each domain, producing an ever-shifting series of state combinations. These shifts are called *reversals*. We reverse between states in a domain based upon the meaning we attach to a situation and whether our values are being fulfilled or not. Figure 1 describes the four domains and the two

Table 1: *Reversal Theory*

Reversal Theory: Key Points
• General theory about what motivates self and others.
• Practical tool for understanding change and your reaction to it.
• Provides a way to recognize emotions and respond in new ways.
• Based on 30 years of research and applied use.

Figure 1: *The Motivational States of Reversal Theory*

Serious	**Conforming**	**Mastery**	**Self**
Future Goals, Achievement	Belonging, Rules	Power, Ability	Self-Oriented
Values ambition and future focus	Values tradition and duty	Values control and strength	Values self-reliance and own needs
Avoids arousal, risk, and anxiety	Seeks group identity	Seeks competence and pride	Takes personal responsibility

MEANS-ENDS	**RULES**	**TRANSACTIONS**	**RELATIONSHIPS**
Does motivation come from achieving the goals or experiencing the process?	Are rules, traditions, and expectations supportive or restrictive?	Are motives based in power and control, or in care and emotional support?	Are you motivated by fulfilling your own needs or another's?

Playful	**Rebellious**	**Sympathy**	**Other**
Process, Passion, and Fun	Freedom, Change	Relationship, Care	Other-Oriented
Moment driven and present focused	Rules seen as restrictive	Values compassion	Values giving and generosity
Seeks excitement to avoid boredom	Values innovation and change	Seeks personal connection	Focused on others' needs

opposing states in each pair.

The following are some examples of how the states and reversals work in technology environments:

- Mark is a programmer. More often than not, he is driven by goals, time requirements and a desire to achieve (Serious state). Today, however, Mark is fixing a bug, and in a moment of trouble-shooting he is excited by the intrigue of the problem. In the process and passion of the work, time melts away (Playful state).
- Wendy is a test engineer who generally feels comforted and supported by the test routines that she is supposed to follow (Conforming state). Suddenly, however, she realizes the standard approach misses a critical element, and she decides to stand up and advocate for a change (Rebellious state).
- Bob is a requirements analyst who is documenting the workflow of a specific user group. He begins the discussion focused on developing his expertise in the user's business process (Mastery state). Later, when users express concern about having to abandon their existing tools, he instead begins to feel empathy for their position (Sympathy state).
- Karen is a program manager who has been seeking input from others and focusing on their interests (Other state). Today, she stops and considers what she thinks is the right direction for the project on her own (Self state).

Detecting the Need for Different States

All the motivational states offer benefits to both individuals and teams; these benefits can be realized by developing access to and skill with all the states. Is it the right time to advocate leading edge technology to maximize capability (Mastery), or is it time to care about the concerns of other users (Sympathy)? Is it time to follow structured methodologies (Conforming), or to push against the status quo to innovate (Rebellious)?

Recognizing what is working in the moment can be helped by analyzing patterns over time – states underused in the long term may also need to be accessed more in specific moments. Table 2 outlines what a team *looks like* when states are working well, and it lists indicators that suggest the state might be needed more – both in an individual and in a team [2].

When State Is Working Well	Signs a Person or Team May Need This State
Serious – The Serious state is focused, ambitious and time-aware, driven to make progress and achieve goals – attaching significance to the outcomes at stake.	• A lack of driving vision or goals; people do not see or take seriously the risks that face them, or consider how today's work impacts the future. • Low follow through on projects or commitments, wasted time, money, or other resources.
Playful – The Playful state is enthusiastic, adventurous, and open to risk. Attaching no significance to their actions or activities, people are free to be joyous and fully in the moment.	• Lack of passion and enthusiasm, people who are anxiety prone or overly stressed. Laughter and fun are not frequent or consistent elements of group or work life. • Difficulty knowing where to start on a project because of all the implications associated with every choice.
Conforming – The Conforming state has a clear sense of duty and order and a unifying group identity, within which both trust and the approval of others are prized.	• Mavericks that cause friction and needlessly reinvent the wheel, rather than finding out what has worked in the past. • Little concern for tradition or people's expectations, loyalties and commitments. Need for autonomy has impaired teamwork and unity.
Rebellious – The Rebellious state is innovative and original, pushing against authority and expectations and sparking change.	• Over-commitment to routine, policy, and tradition or missing the chance to innovate or improve. Thinking is limited to what is known and deemed appropriate. • Discomfort with change – tend to apply incremental solutions where fundamentally different approaches may be needed.
Mastery – The Mastery state is motivated to be confident and competent, focusing on gaining skills and power to be used for an individual's own benefit or someone else's.	• People lack power, competence, problem-solving skills, and the ability to exercise control over events, tasks, and action. • Hyper-sensitivity to conflict – frequent needs for approval, affirmation and affection.
Sympathy – The Sympathy state is interpersonally attuned, sensitive to the feelings of others, and both gives and asks for emotional support.	• Underdeveloped personal and professional relationships; blindness to the benefits of trust, friendship, and human connection. • The hunger for power and drive to win trumps personal relationships, values, and emotional considerations.
Self – The Self state is self-aware, responsible, and individually accountable.	• Avoidance of personal responsibility or accountability. • A tendency to rely on others to solve problems rather than solve them alone.
Other – The Other state is unselfish and motivated to be part of something that transcends themselves and individual concerns.	• Loners who miss out on team spirit, community, and connection to meaningful causes. • Strong focus on individual achievement or need leads to appearance of disinterest in the needs of external interests or the group overall.

Table 2: *Recognizing When New States and Emotions Are Needed*

Accessing Different States: Creating Reversals

So far, we have concentrated on recognizing the motivational states and associated emotions, as well as described how the states play out. Now, we turn to action planning. There are a myriad of actions available to trigger different states in an individual or a team. Here are examples that have helped technology and systems development teams we have worked with.

Triggering the Serious State

The motive in triggering the Serious state is to increase the emphasis on goal achievement. One way to do this is to introduce or enforce formal development methodologies, work breakdown structures, and risk management. (Any imposition of managerial or group expectation or norm is also accessing the Conforming state; however, this suggestion focuses on goal-orientation, triggering the Serious state.) In addition, talk about examples of what failure might

bring: functional failures, reputation damage, and perhaps even loss of life. What happens when your work is not done well or on time? Talking about the larger or long-term implications of near-term successes can create the Serious state in a team. What future pay-off might be there for work that is done well or on-time?

Triggering the Playful State

Triggering the Playful state is often useful in decreasing anxiety and increasing emphasis on passion and fun. An effective way of doing this is to be enthusiastic and excited about what you are saying and doing – raise your eyebrows and the pitch and tempo of your voice. When a complicated problem emerges, encourage others to just *dive in anywhere* by setting aside an open period to share cool ideas. Many teams suffer from an over-emphasis on the Serious state, so look actively for opportunities to inject fun and enjoyment into meetings and everyday work life: Supply and engage yourself with colored/scented markers, bright paper, fidget toys, puzzles and games, or hold a party for the team or organization to celebrate something or just have fun together.

Triggering the Conforming State

This state offers the benefits of increased team identity, shared expectations of process, and sense of belonging. One way to spark this state is to introduce a capability model to focus attention on the way things should be done. This can then be carried through by having regular meetings where team members are all expected to attend and play a specific role (time-keeper, note taker, etc.). Team identity can also be forged by engaging in ritualistic or ceremonial group events such as awards, review sessions, meetings and the like, or encourage the group to enter a corporate contest, like an intramural baseball tournament or a volunteer event together. By talking about it before and after the event, you can further encourage participation and belonging.

Triggering the Rebellious State

To help increase feelings of freedom and independence, challenge someone or a group by suggesting that a given action or achievement is not possible. Other actions can include provoking an argument or healthy debate; criticizing rules, tradition, or some opposing force; asking *why*; and prodding others to do the same. The Rebellious state is often triggered by questioning, testing, and pushing against the established way of doing things – urge your team to seek improvements in established approaches.

Triggering the Mastery State

In this state, the motive is increasing confidence, pride, and ability. One way to do this is to take control of a meeting or conversation by standing up, using both your voice and body to appropriately project conviction, confidence, and competence while coaching those who work with you to do the same when appropriate. This state is also often sparked by challenging and driving yourself or others to craft solutions to tough problems and find answers to complex questions. Finally, one of the best actions from the Mastery state is to teach, mentor, or coach someone to transfer power and ability to another.

Triggering the Sympathy State

We have encountered many teams that would benefit from an increase in empathy and care – both for themselves and for others. One powerful tool for doing this is to create representative *stories* about different user groups, so that developers have a specific person in mind to care about when creating a product. (For more on this tool, see Cooper's reference to personas in [3].) Within the team itself, do not resist telling people personally and face-to-face how much you value them. You would be amazed at the positive impacts that come when you treat your colleagues as you would want your best friend or family to be treated at work.

Triggering the Self State

Increasing feelings of self-reliance and personal responsibility often requires demanding and modeling individual accountability for decisions and actions. Be clear about who *owns* what piece of the project or effort and ask for updates on individual progress. Another path is to speak with your team members or colleagues about their own lives and professional development plans and goals. Outside of the team, what interests them? (Note: This requires that the giver of the action be in the Other state to encourage the receiver to be in the Self state.)

Triggering the Other State

The motive in the Other state is to increase feelings of altruism and transcendentalism. To do this, speak to the team about the mission, the cause, the *larger than life* ideas: calling out faith and motivation and inviting others to join the group that is fighting the good fight. (Note: These actions may also spark the Conforming state if framed as a path to team belonging or the Rebellious state, if the greater good involves fighting another entity, or if defending the rights of an underdog is

present.) Another way to trigger the Other state is to ask the team to role play or brainstorm what their client or customer is thinking. Set aside some time for *walking in their shoes* and determine how the group can meet those needs.

Implementing Reversal Theory in Teams

Now that we have reviewed ways to detect the states and possible actions to trigger different states, let us outline a structure and process for implementing this tool to help build and unify a team.

Step 1: Gather Information – What's Going On?

Before taking action, ask what is going on with the team. Can you match what you see to a motivational state? Is the team generally overly stressed or rules bound (Serious or Conforming)? Do they just talk about work, with little personal connection (Mastery)? Is, instead, the team experiencing conflict because different states are working in opposition? For example, do those who want the team to *lighten up* (Playful) conflict with those who want the team to *tighten up* (Serious)? Use the structure of the states to analyze team behavior and develop hypotheses for what is going on.

Step 2: Identify Target – What Do You Want to See?

Ask yourself next what you are trying to do. Does the team need more clarity of purpose (Serious)? More creativity (Playful, Rebellious)? More group unity and shared vision (Conforming, Other)? Are you instead interested in helping team members recognize the value in different states to reframe an existing conflict? Review the benefits offered by each state described previously. Which state(s) do you seek to create in your team?

Step 3: Select and Take Action – What Changes?

Many actions can trigger a state that might benefit the team. You can also work to change the situation, so the states experienced are more positive (e.g., for a team that is anxious in the Serious state, work to postpone a deadline providing room to breathe). In this step, therefore, select and then take an action and see if it creates the change you want to see. Three important words of caution: Authenticity is key. If your team members perceive your actions as manipulative or fake, they may feel distrust and cynicism. As such, if you want your team to be in a certain state, try to

trigger that state in yourself first. It may be easier for your team, for example, to access the Playful state if they see you excited about the work at hand.

Step 4: Monitor and Adjust – What Is Next?

Reversal Theory focuses on change and variability – as such, one state change can cause others, both positive and unexpected. A change in the setting can then shift states again; pushing one state lever can cause others to shift as well. As such, the final step is to continue to monitor what's going on and then repeat these four steps, adjusting actions to both lead change and respond to emerging emotions and needs.

Conclusion

Motives underlie all human action and can lead to a wide range of emotions – both positive and negative. In our experience, effective leaders and teams do the following:

- Know their own motivational tendencies and the impact they have on others.
- Know the state most needed by individuals, teams, and organizations at any given time.
- Trigger specific states in individuals, teams, and organizations as needed.
- Model and encourage motivational diversity.

We frequently recall a deputy chief information officer comment a few years ago when urging transformational change across the Department of Defense: *Change begins with the people doing the work,* she said [4]. We agree. Change begins with each of us: one state, one choice, and one emotion at a time.◆

References

1. Apter, M.J. "Reversal Theory: The Dynamics of Motivation, Emotion and Personality." Oneworld Publications, 2007.
2. Rutledge, H., and J. Tucker. "Reversing Forward: A Practical Guide to Reversal Theory." Otto Kroeger Associates (OKA), 2007.
3. Cooper, A. "The Inmates Are Running the Asylum." Indianapolis: Sams, 2004.
4. Guthrie, P. Systems and Software Technology Conference. Plenary Session, 2005.

Note

1. See "The Human Dynamics of IT Teams" Feb. 2004 <www.stsc.hill.af.mil/crosstalk/2004/02/0402Tucker.html>, and "Transforming Cultures: A New Approach to Assessing and Improving Technical Programs" Jan. 2006 <www.stsc.hill.af.mil/crosstalk/2006/01/0601RutledgeTucker.html>.

About the Authors

Jennifer Tucker is consulting director at OKA, where she serves as a group facilitator and trainer, management consultant, and project manager. Her most recent work has been with information technology projects and teams – specifically in the public sectors of defense and health. Tucker holds a bachelor's degree in environmental science from Wesleyan University and a master's degree in science in management from Purdue University. She is a certified Project Management Professional and co-author of "Reversing Forward: A Practical Guide to Reversal Theory."

OKA
3605 Chain Bridge RD
Fairfax, VA 22030
Phone: (703) 591-6284
E-mail: jtucker@typetalk.com

Hile Rutledge is managing partner of OKA. He is an experienced organization development consultant, trainer, and public speaker with a background in management, adult education, and leadership development. Rutledge is co-author of the revised "Type Talk At Work," and lead author for "Reversing Forward: A Practical Guide to Reversal Theory." He has a bachelor's degree in humanities from Hampden-Sydney College and a master's degree in organization development from American University.

OKA
3605 Chain Bridge RD
Fairfax, VA 22030
Phone: (703) 591-6284
E-mail: hrutledge@typetalk.com

The Gauge That Pays:
Project Navigation and Team Building

Kasey Thompson
Software Technology Support Center

Tim Border
Self Management Systems

Managers can build better, more efficient teams and successfully navigate the toughest project environments in tandem by reading a set of human gauges (indicators) provided by their project team. Reading these gauges requires observation of, and listening to, some rarely utilized aspects of teaming. Once collected, the gauges act as essential leading indicators that provide insight into building stronger teams and arriving at a project destination safely and on time.

Teaming is a fragile thing and no one person or organization has the market cornered. Some do it well on a more consistent basis, but all have difficulties. The question is, *why is teaming so difficult and how can we regularly get better?* We might find some answers in the following story of how an old navigator taught a lesson on teamwork as related by the co-author of this article, Tim Border:

An older gentleman who was squeezed into an old Air Force uniform stood next to me while in line at the airport. I felt compelled to thank him for his service to our country, and he humbly apologized for the way he looked in his old uniform, explaining that he had been asked to participate at a World War II convention.

Coincidentally, we were seated next to each other on the plane and picked up our conversation where we left off. He began to tell me a story of when he was a young navigator, and he felt green along side the more seasoned pilots on his very first mission.

They headed for their destination with a team of three inside the cockpit and many more supporting them on the ground. Watching the instruments closely, he noticed they were slightly off course. He explained to me that if you are one degree off course and fly that way for one hour you will be one mile off course. He kept waiting for the pilots to make the correction. Suddenly, almost without warning, enemy aircraft were firing on their plane. It happened so fast, he knew they were not going to make it.

Both the pilot and co-pilot were killed in the crash.

"I didn't say anything to them about being off course. I thought they knew. They had to know they were in enemy territory. If only I could do it again. I would boldly

state to my ranking officers we were off course. It was my job and I didn't say anything. I am still haunted by the memory. As a navigator, I am responsible to the team to keep us flying on course." He continued, "Eighty percent of any given flight is off course, from ascending, moving through traffic, storms, and the descent. Flying requires constant monitoring and adjusting. Our instruments are critical in helping us navigate course challenges, and we consider them to be vital in providing accurate, timely information so we have the awareness to make necessary changes if needed."

He told me he has spent a lifetime telling that story hoping individuals and teams would learn from his mistakes and listed the following principles as successful keys:

- Know where you are going.
- Be aware of what could take you off course.
- Make the necessary course corrections as soon as possible when off course.
- Execute with integrity and communicate continually.

We now apply and share his wisdom in regard to building, maintaining, and most importantly, guiding successful teams.

Hard and Soft Gauges: Traditional Versus Human Indicators

Much like the navigator needed instrumentation, we need to choose the correct gauges to keep our team on the road to success. Some common and useful *hard gauges* are cost, quality, and schedule. These are valuable for showing managers they are off track and are easily measurable. However the information indicates very little about what actually happened in the process. The information provided by

these gauges is not enough to ascertain where things went wrong and, more importantly, why things went wrong. These hard gauges, while useful, leave managers wanting for additional information to make decisions on how to efficiently correct what has gone wrong. This wanting hints at the need for additional gauges that provide more leading indicators of early warnings and paint a more complete picture of reality when combined with the hard gauges.

It would be nice to have a single warning light that burned red at the moment projects go astray. Unfortunately, that gauge does not exist, but we may have gauges that can produce the same warning if astutely measured and monitored.

So what is the gauge that pays? It is the human gauge or *soft gauge* – otherwise known as the individual members of our project teams. It becomes essential in times of project peril to speak with the people performing the work. Team members act as the soft gauges, or early warning systems. Some useful soft gauges are *relational conflict*, *communications,* and *functional conflict* (see Figure 1). Managers, with the use of soft gauges, can pinpoint a desired destination and correct the team's course when combined and contrasted with hard gauge readouts. Soft gauges provide added information to arriving safely and on time, all while building a highly dynamic team en route to success.

This article includes soft gauge information regarding *reading the gauge, making course corrections*, and *performing observational insights* (OI). The OI section reflects observations of a team formed four years ago as it struggled, grew, and overcame obstacles through the use of soft gauges.

Relational Conflict Gauge: How Far Off Track Are We?

Teams will always have an intrinsic flaw that can potentially derail progress. That flaw resides in the fact that they are composed of members of the human race. One significant difficulty we humans have is the sometimes conflictive ways in which

we interact, communicate, and function in an interdependent environment. The resultant conflict contained in team interaction is a powerfully destructive force, but it is also a source of unification when used correctly.

There is an old riddle that asks, *what can blind a man yet make him see? What erodes mountains yet constructs buildings?* The answer is sand, as it is a fundamental element in the making of glass (spectacles/eyeglasses) and brick (the building blocks of many an edifice). Sand also blinds those poor souls who are trapped in a desert storm, and wears down mountains and structures through erosion. The answer could very well have been *conflict* as it possesses the same qualities as sand. One component of conflict can destroy relationships, bonds, and synergy when attempting to build teams. But, when used correctly, there is an underutilized component of conflict that unites, clarifies, focuses, and moves teams forward. Think of conflict as a tool. This tool, like most others, can be used for destruction or construction. Stephen P. Robbins [1] describes the destructive components of conflict as *relational* and the constructive component as *functional*. Monitoring and utilizing these two types of conflict are essential in building an effective team.

Relatively little time is needed to understand the use of relational conflict as a gauge. Relational conflict resides in emotional differences that have little to do with process, purpose, or function. These emotions trigger the *fight* or *flight* response in the brain, draining blood away from the higher functioning portion called the neocortex. Reasoning and decision making are then sent to the brain stem, sometimes referred to as the *reptilian brain*, where reasoning capabilities are significantly reduced. Such emotional differences might include conflict over personal styles, choices, and work habits. Relational conflict often may stem from issues unrelated to work or even social or political issues, stall the project team's creative engines as blaming begins and sides of issues are chosen, and cause the team to fracture, decompose, and reduce overall proficiencies. This decomposition is usually the result of either perceived or actual lack of respect for fellow teammates. As quoted in [2], "Respect is like air. We don't think about it until it's gone – once it's gone, it's all we think about."

Like an engine temperature gauge, relational conflict requires constant monitoring as team differences inevitably arise. Observation and awareness are the
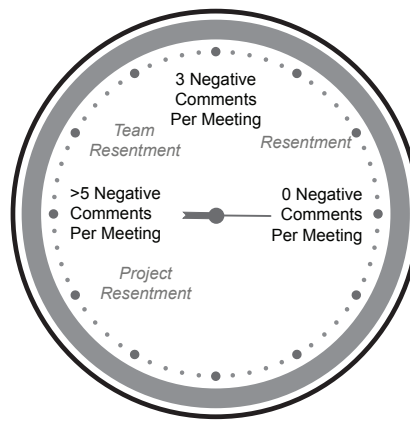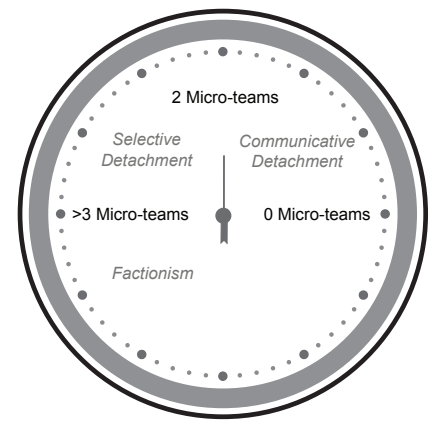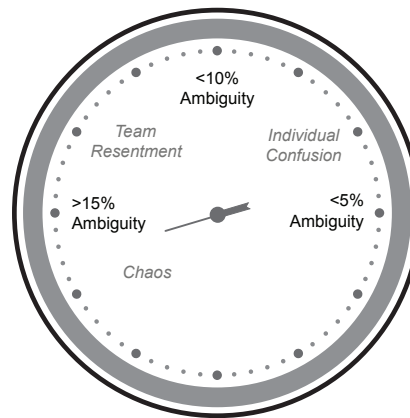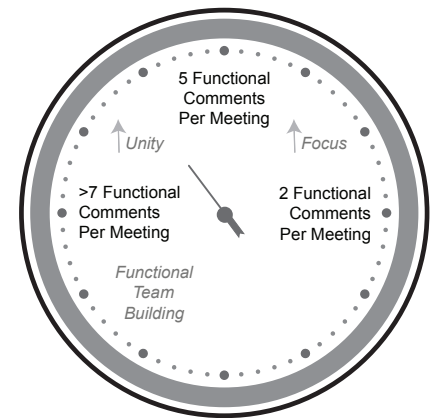
Figure 1: *Soft Gauges*

first steps in any course correction, and managers need only to begin to take and make note of team behavior. Conflict is not always easy to spot. In most cases, conflict begins with subtleties that commonly go unnoticed. Such subtleties grow into larger conflicts, and by that time, the damage has been done and team chemistry has broken down. It is suggested that managers make unobtrusive *fly-on-the-wall* type observations so as not to intrude on the work performed or create artificial behaviors from the team and then witness authentic team dynamics and behaviors.

The concept of conflict appears rudimentary on the surface, but the real understanding of conflict occurs in recognizing the ramifications stemming from conflicting groups and individuals. The actual extent and cost of conflict is difficult to assess, but managers can pinpoint the source of conflict through monitoring team interactions and note the point at which teams become derailed when observed with a watchful eye. Allowing, or merely coping with, conflict is too damaging and expensive to permit. Simply addressing the conflict without understanding the very nature of the issue is dangerous and can compound prob-

lems. Observations noting significant breakdowns in communication need to be taken during team meetings or other team interactions on weekly intervals. Managers can look at the interaction amongst the team to decipher if relational conflict is present and to what degree it has affected the team. Relational conflict typically causes team decomposition in the following three stages:

- The first stage is *communicative detachment*, where people become unwilling to constructively communicate with one another due to the belief that a person is doing things incorrectly or that they have been personally wronged. Individuals now need to prove themselves right by taking a hard stand against the other(s) in defense of their point-of-view. Managers may notice a reduction in lines of communication or posturing and position-taking as overall willingness to work together dissipates. Negotiations and solution-finding are stymied until the positional impasse is resolved.

- The second stage is *selective detachment*. Here, alliances are formed based on the individual team member's views of who is right and wrong in respect to

the disagreement. Selective *micro-teams* are formed within the greater team at large. Micro-teams then informally prepare strategies for their success at the exclusion of the other remaining team members.

- The third and final stage *factionism* refers to the state of functionality of a decomposing team. During factionism, multiple teams function where a singular team once existed. Effort is duplicated and confusion insidiously increases. The collective intellectual power is hampered and team proficiency levels drop. Micro-teams begin to establish new, ad-hoc processes as to how to perform work in a new and dysfunctional environment. Individual cultures are formed. Incidentally, factionism cultures breed future factions until one team possibly becomes many warring individuals battling in personal isolation.

It is helpful to understand this systematic decomposition of typical, dysfunctional teams to better understand the depth of the dysfunction, and to further recognize how to avoid or diffuse relational conflict in our own teams.

Reading the Relational Conflict Gauge

1. Record and retain the frequency of negative personal or team comments made in team meetings.
   * Note: An increase in negative comments is a lead indicator of impending team decomposition.
2. Chart what members are on which side of the issue but do not take a position. Categorize the depth of decomposition in terms of communicative detachment, selective detachment, or factionism via the previous definitions. Take note of the issue that is driving the disagreement so it can be re-addressed later during the functional conflict stage.

Making Course Corrections

Begin to listen and gather (document) points-of-view from both sides. Make every attempt to understand each side of the story without agreeing with or aligning yourself with one side or the other.

Observational Insights

The observed team cycled through each of the three stages of team decomposition three times in their four years together. Each cycle began with a singular act of communicative detachment between two members.

Communication Gauge: Communication Saturation and Understanding

The communication gauge is read by individually asking team members about the purpose of what they do, why this project exists, and what success looks like for the purpose of team understanding. Projects are guaranteed to be off-track if any number of team members cannot answer these questions in a consistent manner. Frequently, the goal of a project is misunderstood. A survey of more than 700 employees and first-line managers from various fields taken during the last four years reveals that only one of six employees feel they received adequate initial communications regarding the purpose and direction of the project on which they worked [3]. The survey also indicates that only one in nine employees received ongoing clarification regarding project purpose and direction. The survey reveals an ongoing need for managers to discover what points of a project are misconstrued and then clarify in order for the team to better understand the overall purpose of the project execution. The goal of this gauge is to reach a level called *communication saturation*, meaning, every person on the team possesses all the information and understanding needed to do their job effectively and in concert with other team members. Communication saturation among team members includes knowing the purpose or goal of a project, the interconnectivity of their individual tasks with those both upstream and downstream, a vivid description of a successful project outcome, and how progress will be measured. Such communication facilitates team empowerment and assists managers in providing guidance that is most needed.

Franklin Covey Organizational Solutions reports that, on average, only 15 percent of employees can correctly list their companies' top three goals, and only 12 percent can ascertain how well they are doing in regard to those goals [4].

Dr. Randall W. Jensen, noted cost estimator, stated the following:

> Software development is the most communication intensive of all engineering processes. This unique software process characteristic suggests significant productivity gains are more likely to be realized through communica-

tion improvement, rather than through technology. Communication effectiveness is a people issue controlled by organization structure, management approach, and the development environment. [5]

Communication among team members becomes increasingly critical on software teams due to the interconnectivity of software, hardware, systems, networks, and most importantly, the people attempting to manage the relationship of them all.

Reading the Communication Gauge

1. On a regular but not too frequent basis (quarterly is suggested), gather input from the team on the following questions:
   a. What is the goal or the purpose of the project?
   b. How does your job relate to others in the process?
   c. What does a successful project look like?
   d. How are we measuring progress?
   e. What would you do differently to improve the process?
   * Note: These questions are also useful as part of an annual performance review to clarify and align goals and measurements. Maintaining safety during reviews is best achieved when the manager approaches team members in an authentic spirit of helping and for the purpose of better understanding by all parties.
2. Concurrently ask team members to describe how they directly add to the success of the project and how they are measuring success.
   * Note: Successful projects require all members of a team to be vividly clear regarding the answers to the above questions. One or two unclear or unfocused team members are lead indicators of future confusion. Remember: confusion breeds confusion.

Making Course Corrections

1. Listen for ambiguity and misdirection in the individual's response, then clarify and redirect energies. Be open and allow input on how individuals will add to and measure future success.
2. Communicate the larger vision and specify detailed instructions where

needed. This is an opportunity for managers to clearly convey their goals and specific needs in relation to larger organizational goals while gathering fresh ideas from the team.

3. Retain individual ideas from the team for future proposals during the functional conflict phase, as it will require newly proposed ideas for constructive discussions.

4. Together with the team member, redefine the description of the team member's role in alignment with the project goal, and document in it the team charter.

Observational Insights

The observed team regularly held meetings where input from individuals was gathered and used in team strategies. Redundancies were removed and responsibilities were better understood by the entire team which led to better overall team efficiency. There still remains some ambiguity regarding individual roles, but confusion regarding who does what and why was significantly reduced.

The team also held a meeting to create a team charter where the previous questions were addressed. Each member defined their own role, provided input to team goals and purposes, and then cross-checked the document with management to assure alignment. The team charter has since been used many times to provide team guidance regarding process and procedures.

Functional Conflict Gauge: Course Correction (Getting Back On Track)

The useful portion of conflict is functional conflict, and it is an effective means of getting teams going in the right direction again. Functional conflict is a disagreement regarding how to do the job, the goal of the project, which processes to use, or which methodology to employ, and it is devoid of emotion. Decomposing teams can use focused functional discussions to regain unity. Factionism stage teams are in need of a compelling and bonding force to marshal its efforts and move in a common direction. Functional conflict provides that force by offering some common ground. That common ground is found in generally held beliefs such as the importance of quality or customer satisfaction. Team discussions need to begin somewhere and high-level topics such as quality or customer satisfaction are

normally safe launching areas to more in depth discussions. Managers can drill-down to more polarizing and specifics issues once safety to discuss these subjects without blaming is restored.

Safety is fragile and is best achieved by speaking in terms of facts as opposed to sharing opinions or stories. Crucial Conversations [2] suggests restoring safety by following the *STATE* methodology:

- **S**hare your facts. Begin by speaking solely in terms of facts to build a foundation.
- **T**ell your story. Explain how you see what has happened in the past.
- **A**sk for others' opinions. Ask others how they see the situation.
- **T**alk tentatively. Do not overstate things or draw conclusions.
- **E**ncourage Testing. Ask others to find flaws in your story.

> *"Managers can drill-down to more polarizing and specific issues once safety to discuss these subjects without blaming is restored."*

The STATE methodology is easy to apply and constructive when used as rules for team discussions. Each member of the team is given the opportunity to share their viewpoint on the issues at hand. Using STATE produces safety that in turn produces more accurate, untainted reporting. It is here where managers gather essential cause and effect data that truly indicates where problems with budget, schedule, and quality originated. STATE limits the emotional flare-ups caused by relational conflict and refocuses discussions to functional topics. Functional conflict unites teams by focusing their attention and energies back on solving problems and accomplishing tasks rather than justifying positions in a conflict. Participants in functional conflict require their brains to use cognitive reasoning skills to discuss facts and solutions via the neocortex as well as other higher capacities of the larger cerebral cortex. Blood flow returns to this indis-

pensable grey matter as individuals begin complex problem solving or rationally addressing possible strategies for achieving goals without derailing emotions present. The human brain is less likely to focus on relational conflict issues because it is limited to cognitively focusing on a singular issue. Functionally thinking individuals are now more apt to reason with each other and combine to find the best answer for the purpose of team success or solving a difficult problem as opposed to proving one side right or wrong. The environment becomes one of safety where people can disagree with each other respectfully without fear of offense. This process reunites teams as they explore potential solutions together. Managers monitor this gauge by embedding themselves in team discussions, observing discussion topics, and noting the nature of the discussions. This is the time to share the team's ideas collected from discussions held during the reading of the communication gauge (communication gauge – making course corrections No. 3). Managers assist team members to stay on task by refocusing discussions on function, not personal stories or opinions.

Robbins states the following:

> Conflict is constructive when it improves the quality of decisions, stimulates creativity and innovation, encourages interest and curiosity among group members, provides the medium through which problems can be aired and tensions released. [1]

Re-stated, functional conflict is the method for engagement in creative problem solving and inquisitiveness. Robbins' studies indicate that, at the end of this process, a team will be more cohesive and have more robust processes, products, and services.

Reading the Functional Conflict Gauge

1. During team interactions, chart the frequency of functional comments such as commentary regarding process, procedures, measurements, objectives, goals, and proposed solutions to problems.
   * Note: An increase of functional comments indicates an increase in team health, ingenuity, clarity, creativity, and focus.
2. Review the charts made during the

relational conflict stage (reading the relational conflict gauge #2) to distinguish if micro-teams are dissolving back into the team at large. Be aware that more effort and more focus on functional matters will be required to reunite teams depending upon the depth of decomposition, i.e., communicative detachment, selective detachment, or factionism.

Making Course Corrections

1. Set team meeting ground rules to use the STATE methodology.
2. Actively redirect and refocus team members on functional matters only.
3. Begin team functional discussions using safe, high-level topics.
4. Move into fore volatile issues only after safety is restored.
5. Empower the team by infusing collected team ideas (communication gauge – making course corrections #3) after reunification has begun.
6. Implement team decisions and watch them go.

Observational Insights

The observed team hit their low point in year three. A specific and successful functional conflict meeting was held in an effort to build back up from the factionism stage. The meeting was governed by ground rules where no opinions/stories could be told. All dialogue was pointed toward solutions and facts. As a result, the team recovered and began to strengthen both socially and functionally. Later, the team discovered and implemented STATE methodologies as part of their culture.

Since that meeting, the team has suffered three additional occurrences of communicative detachment and has yet to decompose into selective detachment.

Currently the team is beginning to demonstrate characteristics of a highly dynamic team such as increased communications, sociability, forthrightness, utilization of individual's diverse strengths, and a willingness to support other members.

Conclusion – Putting It All Together

The information needed to apply the four principles of a successful mission offered by the navigator (know where you are, be aware of what could take you off course, make necessary corrections, execute with integrity and communicate continually) are sometimes hidden in our traditional hard gauges, but it is more apparent and readily available in our soft gauges. We only need to be willing to observe and listen. Longevity and good team health will result from a disciplined and measurable approach to the following:

• Observed relational conflict.
• Increased communications.
• Focused functional discussions.

Teams will always flounder and fail to some degree, but learning the cause and making the correction at an early juncture will save time and money. Soft gauges provide the earliest warning system when monitored correctly, iteratively, and diligently. Soft gauges provide information for managers to know how to successfully navigate project paths and allow their teams to arrive at their destinations on time and intact. Team members, like the navigator, are invaluable and possess a need to feel safe in providing what they view as pertinent information to those who can act upon it. The navigator learned the lesson that team success is reliant upon one another the hard way. Tom Demarco states, "An individual can only succeed to the extent that the whole prospers. And the whole can only prosper to the extent that everyone does well" [6]. Managers cannot be successful unless their team is successful and monitoring these soft gauges will confirm that the human gauge is the one that pays.◆

References

1. Robbins. S.P. <u>The Truth About Managing People ... And Nothing But the Truth</u>. Upper Saddle River, NJ: Prentice Hall, 2003.
2. Patterson, K., J. Grenny, R. McMillan, and A. Switzler. <u>Crucial Conversations, Tools for Talking When Stakes Are High</u>. New York: McGraw-Hill, 2002.
3. Leadership Development Training Program. "Hill AFB Historical Assessment Survey, Cultural Transformation Team." 517 Hill AFB, UT: SMXS, 2003-2007.
4. Harris Interactive. "Execution Quotient Data." Proc. from Franklin Covey Workshop. Salt Lake City, UT, 2005.
5. Jensen, Randall W. "Management Impact on Software Cost and Schedule." CROSSTALK July, 1996.
6. Demarco, T., and T. Lister. <u>Peopleware: Productive Projects and Teams</u>. 2nd ed. Dorset House, 1999.

About the Authors

Kasey Thompson is a coach, consultant, and instructor for the Cultural Transformation Team and the Leadership Development Training Program at Hill Air Force Base, Utah. He also works as a coach and keynote speaker for Human Investments, a process improvement consulting firm, improving results through better relations for corporations, families, and married couples. Thompson has a bachelor's degree in Lifestyle Management from Utah State/Weber State University, a masters of business administration from the University of Phoenix, and is an Arbinger Institute-trained coach.

Software Technology Support Center
6022 Fir AVE
BLDG 1238
Hill AFB, UT 84056
Phone: (801) 586-1037
Fax: (801) 777-8069
E-mail: kasey.thompson
        @hill.af.mil

Tim Border has made a significant difference in the lives of thousands of people throughout the nation. From students to corporate executives, he has helped individuals, leaders and teams find the power within themselves to shed victimization and achieve mastery levels of performance. Border has creatively aligned the principles of personal effectiveness into a powerful process. His entertaining approach makes the journey to success enjoyable. Founder of Self Management Systems, Border is a national keynote speaker, university professor, author, and songwriter. He has a graduate degree in training and development.

Self Management Systems
1815 E 1750 N
Layton, UT 84040
Phone: (801) 544-2272
E-mail: timborder@hotmail.com

Why Should I Use the People CMM?

Margaret Kulpa
AgileDigm, Incorporated

Has your organization started a process improvement effort only to have it stall, or even worse, fail? Are you having trouble attracting and keeping the right employees? Are you seeing organizational skills walk out your door? Perhaps you have just finished a successful Capability Maturity Model® (CMM®) Integration (CMMI®) effort and are wondering, What's next? *Then maybe you should try using the People CMM.*

In the old Introduction to the CMM class, there is an interesting graphic (Figure 1). The figure consists of a three-legged stool. The seat represents the organization. One leg represents Technology, one leg represents Process, and one leg represents People. What the graphic shows is that, in order to have a stable organization, all three legs must be present. Without one or more of them, the stool topples over. The CMM for Software, and now the CMMI series, address the process concerns. Various methodologies (for example, Agile development, Microsoft certification, object-oriented design and development) address technology. But the one area often left unaddressed by organizations is people.

What Is the People CMM?

The People CMM was written to address the need to integrate effective people practices with process and technology. It is a staged maturity model that begins with a basic set of practices and advances through ensuing stages of sophistication and maturity (Table 1) [1]. The People CMM has been around in an earlier version since 1995 and was updated in 2002 based on best practices gathered from practical application in organizations. Although the model was originally written for the problems facing the software industry, the focus has now been expanded to any organization that depends on people to accomplish work – and *that* should be just about everybody.

There are five maturity levels, with each maturity level laying the foundation for the next maturity level. As each level (or stage) is achieved, the capability of the organization to do work, both now and in the future, increases. Each maturity level contains anywhere from three

to seven process areas (PAs). The PAs are a collection of best practices gathered from highly functioning organizations, grouped by a common theme into process categories.

- At Maturity Level 1, there are no PAs. Maturity Level 1 is characterized by chaos and inconsistency. Work is being accomplished, but no one is really sure how. Status, performance, and quality are unpredictable.
- Maturity Level 2 is the managed level. At this level, a disciplined approach (via following the sequence of best practices in the Level 2 PAs) is introduced into basic workforce practices to promote repeatable outcomes. However, each project, unit, or workgroup has its own way of performing tasks.
- Maturity Level 3 is the defined level. This level is characterized by having an organizational way of conducting business. Best practices from units and workgroups established at Maturity Level 2 bubble up to the organizational level, resulting in effective organizational policies and procedures. Managers and workers can tailor this organizational way of doing things as necessary, but the

Figure 1: *Quality Leverage Points*

original organizational process provides some structure and sanity to the way work is done.
- Maturity Levels 4 and 5 provide more autonomy to the workforce and provide *management by the numbers*. Quantitative data are used to align workforce practices with current and future business needs, and to chart a path of improvement that is measurable and highly predictable.

There are also PA threads documented in the model (see Table 2, next page) [2]. These threads show how the PAs are integrated and increase in sophistication as the maturity level increases. The PA threads are the following:
- **Developing competency.** Develop individual capabilities to perform immediate and future work in order to contribute to organizational per-

Table 1: *Stages of People CMM*

Level	Process Areas
5 Optimizing	• Continuous Workforce Innovation • Organizational Performance Alignment • Continuous Capability Improvement
4 Predictable	• Mentoring • Organizational Capability Management • Quantitative Performance Management • Empowered Workgroups • Competency-Based Assets • Competency Integration
3 Defined	• Participatory Culture • Workgroup Development • Competency-Based Practices • Career Development • Competency Development • Workforce Planning • Competency Analysis
2 Managed	• Compensation • Training and Development • Performance Management • Work Environment • Communication and Coordination • Staffing

Levels	People CMM Threads			
	Developing Competency	Building Workgroups and Culture	Motivating and Managing Performance	Shaping the Workforce
5 Optimizing	• Continuous Capability Improvement		• Organizational Performance Alignment	• Continuous Workforce Innovation
4 Predictable	• Mentoring • Competency Based Assets	• Competency Integration • Empowered Workgroups	• Quantitative Performance Management	• Organizational Capability Management
3 Defined	• Competency Development • Competency Analysis	• Workgroup Development • Participatory Culture	• Competency Based Practices • Career Development	• Workforce Planning
2 Managed	• Training and Development	• Communication and Coordination	• Compensation • Performance Management • Work Environment	• Staffing

Table 2: *People CMM Threads*

formance.
- **Building workgroups and culture.** Improve coordination and interaction among individuals and workgroups. (The term *workgroups* has replaced the term *teams* in this version of the People CMM.)
- **Motivating and managing performance.** Measure and develop individual performance; align that performance with organizational objectives.
- **Shaping the workforce.** Evaluate current workforce practices, individual capability and skills, and organizational needs and devise plans to address the gaps.

PA threads may allow organizations to follow an alternate path of improvement. For example, let us say that your organization would prefer to focus on building a truly competent and skilled workforce. There is a PA thread called *Developing Competency*. This thread begins at Level 2 with the PA *Training and Development*. It focuses on preparing an individual to improve his capability to perform his immediate assignments. At Level 3 in the Developing Competency thread, we come to *Competency Analysis*. The purpose of this PA is *to identify the knowledge, skills, and process abilities required to perform the organization's business activities so that they may be developed and used as a basis for workforce practices*. This PA focuses on identifying how various business areas in the organization currently conduct business, defining the processes used, and identifying any commonalities or gaps, in order to fulfill not only current business needs, but future business needs as well. The next PA in this PA thread is *Competency Development*.

Competency Development provides organizational opportunities to workforce personnel to improve their individual capability, and thus, the capability of the organization. As the maturity levels increase, so does the sophistication of the organizational concepts introduced. Other competencies introduced at various levels in various PA threads include mentoring (a formal, structured effort) and empowered workgroups (providing more autonomy to workers, freeing them to perform their tasks with less supervision, and freeing their managers to focus on more strategic business concerns).

It must be remembered, however, that the best way to achieve lasting improvement and organizational change is to implement all the PAs at Level 2 first, and then continue with all of the PAs at Levels 3, 4, and 5. Selecting a path based on PA threads increases the risk of not fully achieving improvement in organizational capability. If you look at the PA discussed as examples in the previous paragraph (Training and Development at Level 2 and Competency Analysis and Competency Development at Level 3), you will discover relationships or links from those Level 3 PAs back to the Level 2 PAs. And each PA that resides in a maturity level also has interdependencies with other PAs in that level. Your organization may decide to select one PA thread to concentrate on, but (because of the interdependencies among the PAs within its own level and outside the levels) you also will have to back up and pull what is needed from the PAs outside the thread you have selected. You also cannot pick PAs willy-nilly. Even when using the PAs thread concept, you must implement the PAs within Level 2 first,

then Level 3, then Level 4, and then Level 5.

So in reality, it is difficult to implement the model via PA threads instead of by PAs within a specific maturity level.

The Most Fundamental Level to Implement (or *Bang for the Buck*)

At Maturity Level 2 – the Managed Level – the People CMM PAs focus on instilling basic discipline into workforce activities to achieve repeatability. This level is the most fundamental to implement, as it is the basic building block for all ensuing levels.

Level 2 consists of six PAs. The PAs at Level 2 are the following:
- **Staffing.** Recruiting, selecting, and transitioning people into, and out of, assignments.
- **Communication and Coordination.** Ensuring timely communication for sharing information and coordinating activities.
- **Work Environment.** Providing physical working conditions and resources to enable work to be performed.
- **Performance Management.** Clear objectives used to measure and improve unit and individual performance.
- **Training and Development.** Ensuring that individuals have the skills required to perform their assignments, with relevant development opportunities provided.
- **Compensation.** Everybody's favorite – remuneration, rewards, and benefits based on contribution and value to the organization.

If you look closely at just the names of the PAs, you will probably draw the conclusion that these are the processes that need to be implemented to provide *incentives* for people to *join* your organization and then, to actually *stay* there. You will also notice that these are the areas that will most likely motivate your employees, offer them career opportunities, and provide them with an infrastructure that supports them in doing their work with the least amount of hassle.

You may also be saying, *Hey – Level 2 looks a lot like my organization's human resources department. I don't work there, so I guess the People CMM is not my problem.* Well, maybe – maybe not. It is true that in the People CMM, the process owners[1] of Maturity level 2 are Human Resources (HR) personnel. But just

because HR owns the Level 2 processes, ownership does not mean that they (HR) are the only ones affected by the processes and are the only ones that have to worry about the processes. HR professionals have stressed that a program based on the People CMM model should not be treated as just an HR initiative [3]. Good People CMM implementation means that individuals, teams, and management share commitment and responsibility [4]. And after all, HR policies are designed for – and affect – the entire workforce.

The remaining three maturity levels contain more advanced practices and basically build on the foundation laid at Level 2. At the higher maturity levels, more people from different areas in the organization get involved in People CMM-based process improvement, process ownership becomes more dispersed throughout the organization (not just HR), and interactions among HR, line management, individuals, and workgroups increase. So, implementing the People CMM is not just an HR effort.

The People CMM as Problem Solver

Looking again at Maturity Level 2, the People CMM can improve an organization's ability to attract, develop, and retain individuals through such PAs as staffing, communication and coordination, and work environment (by establishing an environment that encourages people to join the organization, sharing organizational information of interest to new and prospective employees, and ensuring timely job offers and supportive recruiting practices); performance management and compensation (by appropriately evaluating and rewarding individual performance); and training and development (by motivating personnel by offering ongoing skills development and personal career advancement). Instituting these PAs appropriately makes people want to work in an organization. If people do not want to work in the organization, then they will leave, and the organization's reputation for being a *bad* place to work will leak out to the marketplace.

As part of classes and seminars, the Software Engineering Institute (SEI) collected data on *people* issues that organizations found most worrisome [5]. In addition to the problems discussed above, areas of concern included the following:
- Enabling people to deal with contin-

ual change in the organization.
- Changing organizational culture by moving to a team culture.
- Ensuring consistent communication between management and staff.
- Defining roles and responsibilities.
- Aligning personal goals with organizational goals and business objectives.

Briefly, the People CMM PAs that address these issues are the following [5]:

Enabling people to deal with continual change in the organization. *Staffing* introduces new employees into the organization in an orderly manner,

---

"You cannot expect your employees to do all of the extra work required to participate in process improvement activities perceived to be of no real value to them, or to achieve a formal maturity level ... if your employees feel unappreciated, undervalued, and abused."

Training and Development orients employees to organizational practices, and *Competency Development* integrates employee skills with organizational competencies. Overcoming resistance to change is addressed by practices in *Communication and Coordination* that stresses communicating organizational values (including policies and procedures related to change) and expectations of managers and employees, and *Participatory Culture* empowers employees to suggest organizational improvements and make decisions related to their work.

Changing organizational culture by moving to a team culture (remember, the term *workgroups* replaces the term *teams* in this version of the People CMM). *Communication and Coordination* communicates organizational values regarding workgroups and identifies

dependencies to be coordinated among them. *Participatory Culture* and *Empowered Workgroups* empower workgroups to make decisions regarding the conduct of their work. *Competency-Based Practices* defines process abilities and skills that can be applied to workgroups, and *Workgroup Development* identifies opportunities for establishing workgroups and planning work around those groups.

Ensuring consistent communication between management and staff. *Communication and Coordination* contains practices that encourage the formation of communication mechanisms up, down, and across the organization. *Performance Management* uses the information communicated to effectively monitor and measure individual performance by managers and employees, and *Participatory Culture* uses the information communicated to allow individuals and workgroups to make appropriate decisions related to their work.

Defining roles and responsibilities. Staffing and Competency Analysis analyze the work to be performed, the knowledge, skills, and process abilities needed to perform it, and map roles and responsibilities to the work. Training and Development, Career Development, and Competency Development ensure that staff can perform their assigned work, as required by their roles and responsibilities. Participatory Culture defines who may make decisions under what circumstances.

Aligning personal goals with organizational goals and business objectives. Performance Management defines individual performance objectives. Communication and Coordination provides information about organizational performance to individuals. Performance Management and Participatory Culture provide ongoing feedback to individuals about their performance. Organizational Performance Alignment maps performance results at all levels to individual, workgroup, unit, and organizational goals.

Not only can these issues result in poor workforce performance, they can also cause process improvement efforts underway in organizations to stall or fail. Process improvement requires some level of participation from most of your organization. You cannot expect your employees to do all of the extra work required to participate in process improvement activities perceived to be of no real value to them, or to achieve a formal maturity level rating to keep your organization in business, if your employ-

ees feel unappreciated, undervalued, and abused. People will see that the only real opportunity offered in such an organization is to leave. And they will. It is easier to leave than stay and work in a nasty place that only cares about *building the business* and not building its people.

Conclusion

Why should we use the People CMM? Short answer – using the People CMM provides a structure for unstable organizations to become more stable. Has your quest for a CMMI level rating stalled? Is it in trouble? There are many potential reasons for the problems you are encountering, from lack of management commitment to inadequate resources and funding to the overcoming of resistance to change. Organizations have reported that when their CMM or CMMI efforts ran into trouble, concentrating on the lessons from the People CMM provided enough stability and enough guidance for organizational change to get their process improvement efforts back on track [3, 4].

Other organizations that have been successful in implementing the CMMI continue their process improvement journey by selecting and implementing the People CMM. Based on their success with the CMMI, these organizations are concentrating on supporting their workforce in order to continue successful CMMI practice, and to keep their employees excited about the work they are doing. These organizations see the need for improving the capability, not only of their technical processes, but also of their workforce practices. As such, they are using the People CMM as their guide [6, 7].

A very smart man in one of my classes finished his presentation as follows: *Why should we use the CMM? Because CMM stands for* **C***an* **M***ake* **M***oney.*
I think that says it all.◆

References

1. SEI. Carnegie Mellon University (CMU). "Introduction to the People CMM." People CMM Class. SEI/CMU, 2006.
2. Curtis, Bill, William E. Hefley, and Sally Miller. <u>The People Capability Maturity Model®, Guidelines for Improving the Workforce</u>. Addison-Wesley, 2002.
3. Curtis, Bill, William E. Hefley, and Sally Miller. "Experiences Applying the People Capability Maturity Model." CROSSTALK, Apr. 2003 <www.stsc.hill.af.mil/crosstalk/04/2003>.
4. McWeeney, Jean, Carol Sheehan, and Alison Raffalovich. "A Case Study in the Culture of Process: Synergistic Maturity in P-CMM and SW-CMM." Presentation, 10th Software Engineering Process Group (SEPG) Conference, Mar. 9-12, 1998.
5. Hefley, William E., Sally Miller, and Ira Monarch. "Defining People Issues: Why Individuals Turn to the People CMM for Solutions." Presentation, SEI Symposium, 1999.
6. Subramanyam, V., Deb Sambuddha, Priya Krishnaswamy, and Rituparna Ghosh. "An Integrated Approach to Software Process Improvement at Wipro Technologies: veloci-Q." CMU/SEI-2004-TR-006, ESC-TR-2004-006. Pittsburgh: SEI/CMU, 2004.
7. Vu, John D. "Synergy: A Roadmap for Cultural Transformation." SEPG Conference, 2006.

Note

1. Process ownership may reside with an individual, group, or organization. Process owners coordinate various activities associated with process, such as writing processes, changing processes, ensuring that processes are implemented in an organization, and acting as the designated point of contact for process-related information and activities.

About the Author

Margaret Kulpa is the chief operating officer of AgileDigm, Incorporated, an international process improvement consulting firm. She has an extensive background in systems and software development and process improvement. She is the primary author of "Interpreting the CMMI: A Process Improvement Approach." AgileDigm, Incorporated is an SEI partner, providing SEI-authorized services for the Introduction to the CMMI, Standard CMMI Assessment Method for Process Improvement appraisals, and the Introduction to the People CMM. For more information, please go to <www.agiledigm.com>.

AgileDigm, Incorporated
11 Twelve Oaks Trail
Ormond Beach, FL 32174
Phone: (386) 673-1384
E-mail: margaret.kulpa
**   @agiledigm.com**

WEB SITES

Project Smart
www.projectsmart.co.uk
Project Smart is the project management resource that helps managers at all levels to improve their performance. They provide an important knowledge base for those involved in managing projects of all kinds. With regular updates, Project Smart keeps you up-to-date with the latest project management ideas.

Joint Program Management Handbook
www.dau.mil/pubs/handbook/handbook.asp
The Joint Program Management Handbook provides a quick guide to assist experienced acquisition professionals assigned to a joint acquisition program. It offers current policy and advice on joint program issues related to service responsibilities, capabilities and requirements, project manager authority, and fund-

ing. The views of experienced joint program managers are highlighted within this guide to give practical advice to the reader. Lessons learned and practical guidelines derived from Joint Working Group deliberations (November 2003) and from Defense Acquisition University faculty, who have joint program management experience, have been included.

Project Management Library
www.managementhelp.org
The library provides easy-to-access, clutter-free, comprehensive resources regarding the leadership and management of yourself, other individuals, groups, and organizations. Content is relevant to the vast majority of people whether they are in large or small for-profit or nonprofit organizations. The library has one of the world's largest collections of these resources.

Tools for Decision Analysis and Resolution

Dr. Richard D. Stutzke (late)
Science Applications International Corporation

Engineers, managers, and estimators must often select the best items from a large list. One example is which features to include in a product or system. This article describes two types of decision-making methods to help do this.

Systems integration projects require that many kinds of decisions be made, as illustrated in Table 1. The evaluation and selection of alternatives is often difficult as multiple individuals may be required to reach a consensus by considering one or more decision factors.

Making any decision requires time and money. Thus, the first decision should be whether the issue merits the use of a formal decision method. If making a wrong decision will have significant impacts for the project, then the team should use a formal decision method. Typical criteria for deciding if a decision is significant are cost, delay, safety, and corporate liability. For example, if a particular component contributes 50 percent to the total cost of the system, then careful analysis is warranted.

This article describes two types of decision-making methods: voting and multiple criteria decision-making. Voting techniques allow a group to rank alternatives based on unstated criteria. Multi-Criteria Decision-Making (MCDM) techniques allow a group of people to specifically identify important criteria and then combine ratings of these criteria to identify the *best* alternative. These methods can easily be implemented using spreadsheets.

Voting Techniques

Voting techniques provide a way to rank a set of alternatives. Political scientists and game theorists have studied many types of voting. Each voting scheme has particular advantages and disadvantages with respect to various criteria such as fairness, monotonousness, sensitivity to minor changes in the way the votes are cast, abstentions, insincerity, and susceptibility to manipulation.

Approval voting allows each voter to vote for (or approve of) as many items as he or she desires; however, each voter can cast only one vote for each item. The item with the most votes wins. (Alternatively, N items having the largest numbers of votes may win.) Approval voting is simple to understand and use. Many governments and organizations around the world use it to elect officials.

Table 2 shows an example of approval voting used to select features from a list of five features, here identified as A through E. There are five voters (stakeholders) whose votes are shown by the Xs in the table. The right-hand column shows the total votes received by each feature. Feature D is the winner, with feature B the second choice.

The bottom row of the table shows the total votes cast by each voter. The number of votes cast varies, perhaps indicating that the stakeholders did not carefully consider their choices and so the result may not truly represent the best consensus. To obtain a *better* result, use techniques that require the voters to make a *standardized commitment* and so improve the sampling of the stakeholders. Two ways to do this are the nominal group technique and multi-voting.

The *Nominal Group Technique* produces a consensus of rankings. Assume that the number of items in the list is L. Allow each person to choose N items, where N is approximately the total number of items desired on the final list. (Choosing a value of N that is less than the final number of items desired will force the stakeholders to make careful decisions.) Each person must select the N most important items, ranking them from N (most important) to 1 least important. (This is a good ranking system, since the items receiving no votes will have a zero score. Scoring would be more complicated if 1 denoted the most important item.) The facilitator

Table 1: *Typical Decisions for Systems Integration Projects*

- Choose product features (with/without other constraints).
- Identify the *best* design option (trade studies).
- Decide whether to make, reuse, or buy.
- Select a Commercial-off-the-Shelf (COTS) component or tool.
- Pick a vendor or subcontractor.
- Select a risk mitigation approach.
- Decide to bid or not to bid.
- Terminate integration testing.
- Modify work products that are already baselined.

Table 2: *Approval Voting Example*

Feature	Stakeholders					Total Votes
	1	2	3	4	5	
A		X				1
B		X	X		X	3
C						0
D	X	X	X	X	X	5
E	X					1
Total Votes By Voter	2	3	2	1	2	

Table 3: *Nominal Group Technique Example*

Feature	Stakeholder					Total Votes
	1	2	3	4	5	
A		1	2		2	5
B	3	2	3	3	3	14
C						0
D	2	3	1	2	1	9
E	1			1		2

November 2007

www.stsc.hill.af.mil **23**

Feature	Stakeholder						Total	Rank
	1	2	3	4	5	6	Votes	
A							0	-
B	3	1	1	1	1		7	1
C			1	1		1	3	3
D		1			2		3	4
E		1	1	1		1	4	2
F							0	-
G						1	1	5
H							0	-
Total Votes Cast	3	3	3	3	3	3		

Table 4: *Example of Multivoting*
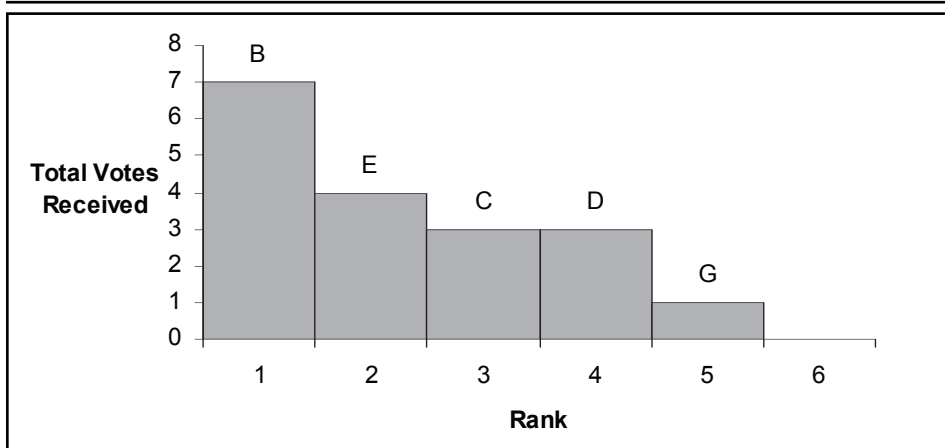
Figure 1: *Pareto Ranking of the Scores*

Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree

Never	Seldom	Sometimes	Often	Always

Hourly	Daily	Weekly	Monthly	Yearly

Table 5: *Some Preference Ratings (Likert Scales)* [1]

Table 6: *Some Preference Ratings (Likert Scales)*

Very Low	≤ four months
Low	> four months and ≤ one year
Nominal	> one year and ≤ three years
High	> three years and ≤ six years
Very High	> six years

collects the votes and totals the votes for each item. The item with the largest total wins.

Table 3 (see previous page) shows an example of five features (A through E) with five stakeholders (1 through 5). Each stakeholder was allowed to choose and rank the three most important items. The last column shows the total score. Feature B is best, with Feature D ranked second.

In *multi-voting*, each stakeholder has a certain number of votes, N, to cast as he or she likes. A stakeholder can cast one, two, or even all N votes for a single item. Table 4 shows an example of eight features (A through H) and six voters (stakeholders). Each voter is given three votes to cast. The table shows the total votes for each feature and ranks the features based on these totals. Feature B is ranked first. Feature E is ranked second. Features C and D have the same total score, but Feature C was selected for third place because three stakeholders voted for it while only two voted for Feature D.

Pareto histograms help one to visualize the strength of the stakeholders' preferences. Figure 1 shows the scores for each item arranged in descending order. Feature B is clearly preferred. Features E, C, and D are close in value. A change of one vote from E to C (or D) would cause C (or D) to move into second place. Feature G is a distant fifth and is only one vote away from obscurity (with Features A, F, and H).

These methods can be used iteratively. The group can vote on a set of alternatives and then eliminate the items having the fewest votes. Optionally, they can discuss the top few items and revise the item descriptions. Then they repeat the process with the revised list.

Techniques for Handling Multiple Criteria

Estimation often involves ranking objects based upon *multiple* criteria. For example, due to limited resources, one might need to choose which set of tasks is the most important to implement, or which design is the best among a set of possible designs.

The stakeholders may also be involved in determining the value or utility of a particular product or process. Often, the value, or benefit, depends on many subjective factors that are difficult to quantify. The academic discipline of decision-making deals with models (normative models) that help associate and measure such factors. The two main types of models are *expected utility theory* and *multi-attribute utility theory* (MAUT).

In *expected utility theory*, the value of a particular outcome is determined by estimating the probability of that outcome and multiplying it by the estimated utility of that outcome. The probability and utility are expressed as real numbers. For example, suppose that a lottery ticket is purchased for $5.00. If the ticket is a winner, the ticket holder will receive $10,000. If the chance of winning the lottery is one in one million, however, the expected return is only $0.01 (= 10^{-6} x 10^{4}). In this case, the expected return does not justify the price of the ticket. Quantitative risk assessment is based on utility theory. Specifically, the risk impact equals the probability of occurrence times the cost of occurrence.

MAUT extends expected utility theory to decisions with multiple alternatives that depend on many attributes or criteria. MAUT maximizes a function of the various criteria, and it assumes that one criterion can counterbalance another (substitution). For example, cost reductions or revenue increases can offset investment or implementation costs. To define a MAUT technique, consider two things: how the properties map to an appropriate measurement scale that pre-

serves the relations between the objects[1] or how orthogonal properties (those with no overlap in what each measures) are selected, and how to define the particular function to maximize. A related issue is defining how to estimate the function's parameters (e.g., the weights that are used to combine ratings for different criteria).

MAUT techniques use preference ratings, illustrated in Table 5, which are ordinal measurements. The ordinal scale only allows comparisons and equality. A restricted ordinal scale that places objects into bins may also be defined. Table 6 illustrates such a scheme. Many parametric estimation models use such rating scales.

To use MAUT do the following:
1. Determine a measurement scale for each attribute, R_i.
2. Map the rating, R_i, to a (ratio scale) value, V_i.
3. Specify the relative weights of the attributes, W_i.

Then compute the score using the weighted sum of the values[2]. For example, suppose that one wants to evaluate four COTS components, named A, B, C, and D. (This example comes from Section 27.5 in [2].) One will want to consider the following factors:
1. Functionality.
2. Integration Effort (person-weeks).
3. Cost (dollars).
4. Vendor Reputation.
5. Product Maturity.
6. Developer Toolkit Available.
7. Training Availability.

The top half of Table 7 lists the rating scale for each factor and its relative weight. (The weights sum to 1.0, but this is not necessary.) The bottom half of Table 7 shows the (ratio scale) values assigned to each rating. Table 8 shows the evaluator's ratings for each factor for each component, with the total weighted score shown in row 6. Component D is the best, with Components A and C nearly tied for second place.

The *Analytic Hierarchy Process (AHP)* is a popular MAUT technique developed by Thomas Saaty [3]. AHP addresses multiple criteria, including subjective criteria. AHP constructs a multi-level hierarchy, such as the one shown in Figure 2. The top level is the decision objective. The bottom level (the leaves) is the possible actions or alternatives. The intermediate levels represent factors that affect the preference or desirability of one alternative, or subfactors that contribute to a factor.

The steps of the AHP process are the

	G	H	I	J	K	L	M
7	**Factor Name**	**Weight**			**Ratings**		
8	Functionality	0.30	VL	LO	NM	HI	VH
9	Integration Effort (person-weeks)	0.20	1	2	3	4	5
10	Cost (dollars)	0.20	below 500	500 to 1000	above 1000		
11	Vendor Reputation	0.10	Poor	Unknown	Good	Excellent	
12	Product Maturity	0.10	VL	LO	NM	HI	VH
13	Developer Toolkit available	0.05	No	Yes			
14	Training availability	0.05	No	Some	Full		
15		0.00					
16							
17	**Factor Name**				**Values**		
18	Functionality		1.00	2.00	3.00	4.00	5.00
19	Integration Effort (person-weeks)		1.00	2.00	3.00	4.00	5.00
20	Cost (dollars)		1.00	0.00	-1.00		
21	Vendor Reputation		-5.00	-3.00	0.00	9.00	
22	Product Maturity		1.00	2.00	3.00	4.00	5.00
23	Developer Toolkit available		0.00	1.00			
24	Training availability		1.00	2.00	3.00		
25							

Table 7: *User-Specified Criteria and Weights*

	A	B	C	D	E
1	**Description:**	Evaluation of Four COTS products			
2	**Prepared By:**	Mary Smith			
3	**Date Prepared:**	14-Oct-04			
4					
5	**Option =**	**A**	**B**	**C**	**D**
6	**Score =**	**1.80**	**1.30**	**1.75**	**2.30**
7	**Factor**				
8	Functionality	HI	NM	LO	VH
9	Integration Effort (person-weeks)	1	2	3	4
10	Cost (dollars)	500 to 1000	below 500	500 to 1000	above 1000
11	Vendor Reputation	Good	Poor	Good	Unknown
12	Product Maturity	NM	LO	HI	NM
13	Developer Toolkit available	No	Yes	No	Yes
14	Training availability	Some	No	Full	Full
15					

Table 8: *Weighted Scores for the COTS Components*

following:
1. Define the Problem.
2. Construct the Hierarchy.
3. Establish Element Comparisons.
4. Calculate Element Priorities.
5. Calculate Overall Priorities.

The decision-maker identifies which factors are important and defines how they influence one another. Then evaluators make pair-wise comparisons at each level, capturing these in *judgment matrices*, one for each criterion or alternative. The rating matrix is triangular since complementarities are assumed. That is, if A >> B then B << A. If A_{ij} denotes the rating of object i relative to object j for attribute A, the AHP model assumes that: $A_{ji} = 1/A_{ij}$. Since an object is identical (equal) to itself, $A_{ii} = 1$. For a single criterion having N alternatives, the evaluator must make $N*(N – 1)/2$ comparisons. Saaty uses a nine-value preference scale. (Other rating scales can be used if desired.)

AHP uses matrix calculations to determine the preference ratios for

Figure 2: *A Hierarchy for Technology Investment*
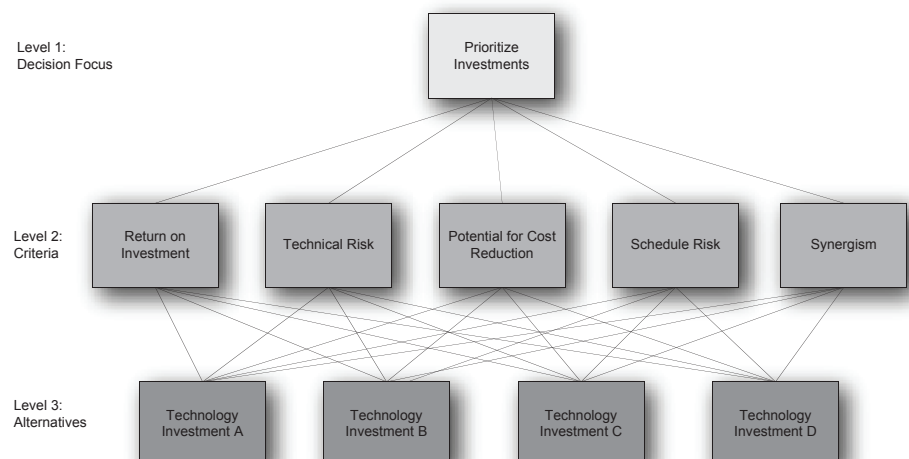
objects and the importance ratios (relative weights) for the criteria at one level (expressed as a priority vector). There are several ways to calculate the priority vector. Saaty computes matrix eigenvalues. Eduardo Miranda uses a geometric mean for a single criterion (software size) in [4]. The calculations in Step 5 usually include consistency tests, allowing planners to revise their relative comparisons to be more realistic (i.e., they repeat Step 4). Then the AHP model combines the relative weights to obtain a ranked list of alternatives (a ratio scale preference vector).

Summary

This article describes simple techniques for ranking and selecting items, as well as ways to evaluate alternatives based on multiple attributes. These techniques all assume rational behavior on the part of the participants. Challenges in applying such techniques in practice are the following:

- People do not always make perfect decisions (due to ignorance, biases, or manipulative strategies).
- People may change their minds.
- There may not be enough resources (time, money) to assign good ratings to all the factors identified.
- Key factors that greatly affect the desirability of the alternatives may not be identified (e.g., the unexpected discovery of an endangered species residing on the project construction site).
- It may be difficult to identify orthogonal criteria.◆

Recommended Reading

For additional information about all of these techniques, including many references, see Chapter 27 of [2].

Part III of [5] is a good place to start when evaluating the economic value of products and systems. Part IIIA deals with cost-effectiveness analysis. Part IIIB discusses multiple-goal decision analysis. Part IIIC discusses uncertainties, risk, and the value of information. Boehm specifically addresses quantities of interest to software and system engineering.

The business community has extensively studied decisions involving financial analysis. Two references that address software-related business decisions are "Return on Software: Maximizing the Return on Your Software Investment" by Steve Tockey [6] and "Making a Business Case: Improvement by the Numbers" by Donald J. Reifer [7].

References

1. Likert, Rensis. "A Technique for the Measurement of Attitudes." <u>Archives of Psychology</u> 140 (1932).
2. Stutzke, Richard D. "Estimating Software-Intensive Systems." Addison-Wesley, 2005.
3. Saaty, Thomas. <u>The Analytic Hierarchy Process</u>. New York: McGraw Hill, 1980.
4. Miranda, Eduardo. "Improving Subjective Estimates Using Paired Comparisons." <u>IEEE Software</u> 18.1 (2001): 87-91.
5. Boehm, Barry W. "Software Engineering Economics." Prentice-Hall, 1981.
6. Tockey, Steve. "Return on Software: Maximizing the Return on Your Software Investment." Addison-Wesley, 2005.
7. Reifer, Donald J. "Making a Business Case: Improvement by the Numbers." Addison-Wesley, 2001.

Notes

1. Measurement assigns directly observed (or estimated) values of some attribute to a mathematical representation that preserves the relationships between the objects in the real world. This guarantees that the mathematical objects can be manipulated and valid conclusions about the corresponding real-world objects drawn. A measurement scale defines a representation and a set of allowed operations on the objects.
2. This is a linear function. This is the usual approach, but the MAUT technique also works with a non-linear objective function.

About the Author

Richard D. Stutzke, Ph.D., was an employee of Science Applications International Corporation (SAIC) and had more than 40 years of experience with software development and project management for scientific, embedded real-time, and commercial systems. Stutzke authored more than 50 papers and articles on software development, estimation, metrics, and management. Stutzke was the principal author of SAIC's software estimating courses and taught more than 1,000 students since developing the course in 1990. His book, "Estimating Software-Intensive Systems: Projects, Products and Processes," won the SAIC Executive Science and Technology Council 2005 Publication Prize Contest in the Technical Book category. The selection criteria included originality of work, significance of results, and effectiveness of presentation. Chapter 27 covers many topics of interest to project planners, systems engineers, and software engineers, including the topics described in this article. For a description of his book, see the book's Web page <http://sw-estimation.com>.

LETTER TO THE EDITOR

Dear CROSSTALK Editor,

I very much liked the theme of your August 2007 issue *Stories of Change* and the change articles. Great topic and one I don't recall seeing before. It was a good reminder that implementing change is not simple but it can be managed successfully. I particularly liked the article *Good News From Iraq*. If change can be successful in that environment, what excuse can the rest of us have for failing?

I would like to add one point about resistance to change, and that is trying to do too much change at one time. People can simply be on overload even if they desire the change. Case in point: In the mid-90s, my base was undergoing a base realignment and closure process, we did a reorganization to a matrix-type organization, and we were downsizing. Three big changes all at once! Any one of them by itself would have been a challenge. A word to the wise: Too much change at once is too confusing, especially if your people are already on overload.

Keep those CROSSTALK issues coming!

– Alan Kaniss
United States Navy
<alan.kaniss@navy.mil>

Common Misconceptions About Service-Oriented Architecture

Grace A. Lewis, Edwin Morris, Dr. Dennis B. Smith, Soumya Simanta, and Lutz Wrage
Software Engineering Institute

Service-Oriented Architecture (SOA) is having a major impact on the acquisition and development of software systems because of its potential for increased business agility, adaptability of applications, interoperability between systems, and reuse of legacy assets. However, organizations often make decisions on SOA adoption without carefully analyzing the implications of their decisions. This article outlines a set of common misconceptions about SOA and suggests ways to more effectively address critical SOA issues that potential users, developers, and acquisition officers may have.

You do not have to look far to become aware of the effect that SOA is having on software systems. Vendors are aggressively marketing hardware, software, tools, and services that support SOA implementation within organizations as diverse as the Department of Defense (DoD), banks, federal agencies, manufacturing companies, and health care providers. Even more significantly, customers are embracing SOA with the goal of reaching a previously unachievable level of interoperability among systems and agility in business practices.

SOA may currently be the best available solution for achieving interoperability and agility, as well as providing a technology upgrade path that preserves the investment in legacy systems and simplifies deployment of new systems. However, our experience from working with customers considering the adoption of SOA suggests that they often have a variety of misconceptions that lead them to greatly underestimate the effort required to successfully implement SOA. These misconceptions are dangerous because they make organizations more susceptible to vendor advertising and hype. In addition, these misconceptions are often embraced by internal IT organizations, leading them to over-promise new capabilities, while underestimating the cost and effort required for achieving even modest improvements. Although some of these common misconceptions also apply to traditional single systems, we focus on their relevance to SOA-based systems.

We hope that by recognizing these misconceptions, organizations can better understand and evaluate the promises of vendors and improve their own internal SOA expectations and planning processes.

Basic SOA Concepts

SOA is a way of designing systems composed of services that are invoked in a standard way. As an architectural style, SOA is neither a system architecture nor a complete system. An SOA-based system is composed of the following:

- Services that are reusable components that represent business or mission tasks, such as customer lookup, weather, sensor placement, account lookup, or credit card validation.
- Service consumers that are clients for the functionality provided by the services, such as end-user applications, systems, or even other services.
- SOA infrastructure that connects service consumers to services.

The most common approach to SOA implementation is that of Web services, which relies on common standards that include HTTP, SOAP, WSDL, and UDDI. However, other SOA-based systems can be implemented using such technologies as MOM, IBM WebSphere MQ, publish-subscribe systems such as JMS, and CORBA.

Some SOA Misconceptions

Seven common misconceptions are identified in the following subsections. The subsection heading represents a statement in the form that an organization might express it. The body of each subsection discusses why the statement expressed in the heading can be misleading. It also provides advice on how to avoid falling into common traps.

SOA Provides the Complete Architecture for a System

Chief among SOA misconceptions is the belief that simply by adopting an SOA strategy for the enterprise, an organization has established a complete well-crafted architecture that will help the organization achieve its IT goals. In reality, SOA is not an architecture, but an architectural pattern from which a number of specific architectures can be derived – both good and bad. An architectural pattern provides guidance to an architect that enables leveraging best practices for that specific pattern. It defines a set of element types, a topological layout of the elements that shows their relationships, semantic constraints on elements, and interaction mechanisms [1]. For example, the elements in the SOA pattern include service consumers, service descriptions, service implementations, and possibly a service bus. One relationship is that between service providers and service consumers. In the case of Web Services, consumers and services are connected by HTTP or HTTPS connectors carrying SOAP messages. Given the architectural elements, or building blocks, any number of systems can be developed based on this architectural pattern. These concrete elements and their interactions are the architecture of the system.

The misconception that SOA provides a complete architecture also leads customers to believe that they can buy SOA *off the shelf*. Although there are a number of products available in the marketplace that can help an enterprise implement

Acronyms	
BPEL	Business Process Execution Language
CORBA	Common Object Request Broker Architecture
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP Secure
IT	Information Technology
JMS	Java Messaging Service
MOM	Message-Oriented Middleware
MQ	Message Queue
OWL-S	Object Window Library for Services
QoS	Quality of Service
SAML	Security Assertion Markup Language
SLA	Service-Level Agreement
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
XML	Extensible Markup Language
WSCL	Web Services Conversation Language
WSDL	Web Service Description Language
WS-I	Web Services Interoperability
UDDI	Universal Description, Discovery and Integration

SOA, none of them are actually an implementation of an SOA-based system. Software architects still need to architect systems based on the SOA architectural pattern. They have to design services and service interactions that meet the qualities that stakeholders expect of the system. In addition, the architect(s) must make decisions on how services are implemented. Service implementations may involve developing new software, wrapping a legacy software system, incorporating services provided by third parties, or a combination of these options.

Information about the quality attributes of SOA-based software systems is just beginning to become available in the literature: One report finds that SOA promotes modifiability, interoperability, and extensibility, but can have a negative impact on security, performance, testability, and auditability [2]. For a given system, the architect needs to understand the quality attribute requirements and needs to architect a concrete system around the tradeoffs that are most important to the stakeholders of the system.

All Legacy Systems Can Be Easily Integrated Into an SOA Environment

One of the most attractive promises of moving towards SOA is that it enables reusing legacy systems, thereby providing a significant return on the investment in these systems. However, migrating legacy systems is neither automatic nor easy. It might not make business or technical sense to migrate the legacy system to an SOA environment.

It is important to understand technical constraints of the legacy components, such as immature technology, that may require significant rework. In addition, it is necessary to understand business issues, such as the business case that will justify the migration of legacy components to services in the specific context. An upfront and hands-on analysis of technical feasibility and the resultant return on investment will help to avoid last-minute surprises.

This analysis must answer at least the following questions:

1. Have consumers for the services been identified?
2. Is it technically feasible to create a service from the legacy system or part of the system?
3. How much would it cost to expose services from the legacy system?
4. What changes will have to be made to legacy systems in order to use these services?

5. How much will these changes affect the current end users of the legacy system and other dependent production systems?
6. Are the costs of exposing services, together with the associated risks of making the required changes, feasible from a business perspective?

The bottom line is that there are issues to take into consideration that go beyond adding a service interface to an existing system.

SOA Is All About Standards and Standards Are All That Is Needed

This statement primarily applies in the context of Web services, the main standards-based technology available today to realize SOA. This leads to a corollary misconception that SOA and Web services are the same. In reality, Web services are only one potential approach to SOA implementation.

It is true that public standards like those supporting Web services are often preferable to proprietary solutions because they are (potentially) supported by a wider community. But, most Web service standards are still emerging and subject to multiple interpretations.

Basic infrastructure standards that support the exchange of messages between service consumer and provider-such as HTTP, XML, XML Schema, SOAP, WSDL, and UDDI are the most developed and mature of the Web service standards. However, being stable for years does not mean that the standards are complete. For example, after adopting basic infrastructure Web service standards, some organizations found that their services still could not communicate information effectively with other services due to different design decisions and flexibility in the standards. The WS-I Basic Profile was constructed to provide better interoperability across implementations using basic infrastructure standards [3]. In addition, revisions to standards are likely in any area undergoing rapid advances in technology.

Standards for service composition (e.g. WSCL, WS-Coordination, BPEL, and cross-cutting standards [e.g. WS-Security, SAML, WS-Transaction, WS-Reliability]) are less mature and far less stable than basic infrastructure standards. Currently, there are a number of competing proposals and standards for service composition and cross-cutting concerns that conflict and overlap. Regarding these less mature areas of Web services, the old saying sums it up – *the best thing about standards is that there are so many to choose from.*

SOA Is All About Technology

Vendors pushing SOA products will (for good reason) promote their technologies as the solution to an organization's IT problems. However, SOA also entails changes to the organization's IT governance model – the set of rules and regulations under which an IT department operates, and the mechanisms to ensure compliance with those rules and regulations. This is especially true if SOA is used to support business processes or mission threads. Therefore, a well-defined governance model that includes items such as the following is essential for the success of SOA implementation:

- Service identification that maps to business or mission goals.
- Service repository management.
- Service implementation guidelines.
- Change management to deployed services.
- Mechanisms, tools, and policies for maintaining and monitoring deployed services.
- Policy enforcement at design and run time.
- Security and access control.
- Definition and enforcement of SLAs between service consumers and providers.

The implementation of SOA in an organization should be part of a larger effort to assure that SOA and related governance are aligned with strategic goals and objectives.

The Use of Standards Guarantees Interoperability in an SOA Environment

True interoperability can only be achieved if service consumers and providers interoperate at both the syntactic and semantic levels. There is interoperability at the syntactic level if they can exchange raw data elements such as text, numbers, or dates. There is interoperability at the semantic level if they understand and agree on the meaning of exchanged data. For example, a spacecraft monitoring application may rely on a service that does an analysis of data received from onboard sensors. The service may correctly perform the analysis of the raw temperature data. However, it may make an assumption that the temperature data is expressed in Celsius as opposed to Fahrenheit. In such a case, there is interoperability at the syntactic level, but not at the semantic level. In this example, both the requesting consumer and the onboard sensor share a common understanding that the number exchanged represents temperature. However, there

must also be a deeper understanding of the meaning of that value, such as the temperature unit or where and how it was measured [4]. The results of an incorrect assumption in this case could prove disastrous for the mission.

In the case of Web services, for service consumers and providers to be interoperable it is not sufficient to agree on the representation of data in XML documents because there is no way to specify the meaning of data in an XML or WSDL document other than in text descriptions. The problem is that text descriptions are imprecise, are often not filled in, and are not readable by machines, rendering them open to multiple interpretations by human developers. Also, even though the full XML Schema Datatypes specification can be used to specify data, it is rare to see anything other than a data type in the WSDL document that describes Web service operations. Optimal methods of describing the meaning of Web service inputs and outputs in a formal manner is still an area of active research [5, 6].

A Service Registry Allows Service Binding Dynamically at Runtime

Currently, binding to services is usually done at design time. This is referred to as static binding or fully-grounded binding. Discovery and composition of services are done at design time such that the developer can discover the syntax and semantics of the service before it is actually used. In the case of dynamic binding, discovery and composition of services are done at runtime. This is currently a complex and poorly supported task.

In a basic scenario of dynamic binding, service consumers retrieve the service address from a registry before each call to the service. If there are several providers of the same service, the service consumer can choose at runtime which one to use. The consumer can also rank providers based on quality of service criteria, choose a preferred provider, and use others as backup if the preferred service is not available.

More advanced automatic discovery and composition of new services at runtime requires the use of common ontologies by service providers and consumers within a domain to describe function and usage of services. Given this shared ontology, it would still be necessary to develop components that can construct the right queries for the discovery of services, compose services when there is not a single service that provides the needed function, and then provide the right data to invoke the discovered service. Current technolo-

gies have not advanced to a point where this is possible in production environments [7].

Testing SOA-Based Systems Is No Different Than Testing Any Other Type of System

Testing service consumers, as well as the services themselves, is challenging for various reasons. Most traditional testing techniques cannot be directly applied to services in the SOA world because testing has to occur at runtime and in real time [8]. Independent testing of a service from a service provider perspective is different from that of a service consumer. Moreover, the service provider and con-

---

"Modeling and simulation can provide some guidance and confidence during the design phase, but they are not a substitution for end-to-end testing of service-based applications."

---

sumer must collaborate and cooperate to ensure correctness and trustworthiness of services [8].

Service consumers can only be fully tested when the invoked services (or test instances of them) are available. The ease of testing will most likely depend on whether the service is internal or external to the organization – there is more control if it is internal.

In an SOA environment it is common for different services to be owned by different organizations and for services to use different technologies. Because an SOA environment is distributed, loosely coupled, and asynchronous, testing can be significantly more complex than simply testing a set of known paths in a single system [9]. Modeling and simulation can provide some guidance and confidence during the design phase, but they are not a substitution for end-to-end testing of service-based applications [9]. Service consumers will necessarily have to be prepared to deal (or not to deal) with degraded service modes and complete service failure.

Services can be reused across applications that cross enterprise boundaries. Changes requested by one service consumer in an existing service can result in undesired results for another service consumer. Changes in service interface and implementation must be tested continuously by each of the service consumers in order to ensure that the actual service behavior conforms to intended behavior. Finally, service providers have to extensively test their services because they cannot anticipate all the possible scenarios in which their service will be used. Testing has to cover functionality, load testing and stress testing, as well as other elements specified in an SLA.

Conclusions

We believe SOA may be the best current approach for achieving critical interoperability, agility, and reusability goals that are common to many organizations. However, we also believe that the difficult reality of building and managing large-scale SOA-based systems often gets lost in the understandable corporate desire for sweeping improvements and the hype of vendors.

Our intent is not to discourage organizations from adopting SOA, but to caution them about some important issues and risks to consider while creating their SOA strategy. Most of these issues are currently active areas of research in the service-oriented computing community. The solutions will require time to mature.◆

References

1. Bass, L., P. Clements, and R. Kazman. Software Architecture in Practice. Addison Wesley, 2003.
2. O'Brien, L., L. Bass, and P. Merson. Quality Attributes and Service-Oriented Architectures. Pittsburgh: Software Engineering Institute (SEI), Carnegie Mellon University (CMU), 2005.
3. Web Services Interoperability Organization. "Basic Profile Version 1.1." 2004. <www.ws-i.org/Profiles/Basic Profile-1.1.html>.
4. Lewis, G., and L. Wrage. "Case Study in COTS Product Integration Using XML." Proc. of the Third International Conference on COTS-Based Software Systems, Redondo Beach, CA, Feb. 1-4, 2004: 41-52.
5. Martin, D. et al. "Bringing Semantics to Web Services: The OWL-S Approach." Proc. of the First International Workshop on Semantic Web Services and Web Process Composi-

tion. July 6-9, 2004, San Diego, CA.

6. Roman, D., et al. "Web Service Modeling Ontology." Applied Ontology 1.1 (2005).

7. Metcalf, C., and G. Lewis. Model Problems in Technologies for Interoperability: OWL Web Ontology Language for Services (OWL-S). Pittsburgh: SEI, CMU, 2006.

8. Tsai, W. T., et al. "Cooperative and Group Testing in Verification of Dynamic Composite Web Services." Computer Software and Applications Conference, Sept. 2004.

9. Acharya, M., et al. "SOA in the Real World-Experiences." Lecture Notes in Computer Science, Volume 3826, Nov. 2005, pp. 437-449.

About the Authors

Grace Lewis is a senior member of the technical staff at SEI where she currently leads the system of systems engineering team within the Integration of Software-Intensive Systems (ISIS) initiative. Her current interests and projects are in SOA, legacy system modernization, and software development life-cycle activities in systems of systems. She has a bachelor's degree in systems engineering and an executive master's of business administration from Icesi University in Cali, Colombia, and a master's degree in software engineering from CMU.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-5851
Fax: (412) 268-5758
E-mail: glewis@sei.cmu.edu

Soumya Simanta is a member of the technical staff at the SEI, where he is part the ISIS initiative. His current research is focused on system of systems engineering, service-oriented architecture, modernization of legacy systems, grid architecture, and evaluation of current technologies that support integration and interoperability between systems. Previously, he did software design and development in the finance and telecom domains. Simanta has a bachelor's degree in electronics engineering from Sambalpur University, and a master's degree in software engineering from CMU.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-7602
Fax: (412) 268-5758
E-mail: ssimanta@sei.cmu.edu

Edwin Morris is a senior member of the technical staff at SEI. He has more than 20 years experience in software, including design and development of embedded real time operating systems and tools, management of technical staff, and support for military, government, and corporate software initiatives. Morris is currently a member of the ISIS initiative.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-5754
Fax: (412) 268-5758
E-mail: ejm@sei.cmu.edu

Lutz Wrage is a senior member of the technical staff at SEI where he is part of the Dynamic Systems program. His research includes work on SOA and systems-of-systems engineering. Before joining the SEI, he worked in the area of Enterprise Resource Planning systems integration and customization. Lutz holds a degree in computer science from the Technical University-Berlin and a master's degree in software engineering from CMU.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-7771
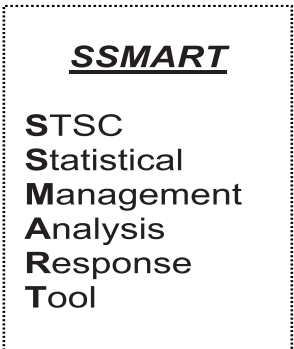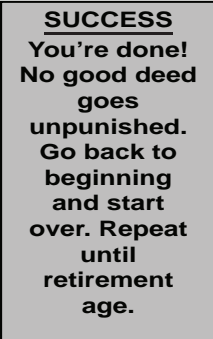**Fax: (412) 268-5758**
E-mail: lwrage@sei.cmu.edu

Dennis Smith, Ph.D., is a senior member of the technical staff and lead of the ISIS initiative at the SEI. This initiative focuses on developing and applying methods, tools, and technologies that enhance the effectiveness of complex networked systems and systems of systems. He has been involved with working with DoD organizations in developing an SOA capability, including issues of SOA strategy, governance and migration of legacy assets to SOA. Previously, he was a member of the Product Line Systems Program and technical lead in the effort for migrating legacy systems to product lines. He has published a variety of books, articles and technical reports, and has given talks and keynotes at conferences and workshops. He was the co-editor of the Institute of Electronics and Electrical Engineers and International Organization for Standardization-recommended practice on computer-aided software engineering adoption, and has been general chair of two international conferences. Smith holds a masters and a doctorate degree from Princeton University, and a bachelor's degree from Columbia University.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-6850
Fax: (412) 268-5758
E-mail: dbs@sei.cmu.edu

SSMART Team Management

Want to skip all of the interesting articles this month? Then use this Software Technology Support Center (STSC) Statistical Management Analysis Response Tool (SSMART) to help you understand how teams work. All you need is a single die. Cut out the tokens, fill in with your team's name, and place your token on START. Roll your die, and advance the number you roll. Follow the directions. One turn per player, one roll per turn.

START Schedule and budget seem plentiful! Roll the die and advance.	**Team now works for matrixed organization. Go back 3.**	**Total team reorganization. Skip 1 turn, then go back to START.**	**Unexpected team absences. Skip 1 turn.**	**Team and management read CROSSTALK regularly. Go to SUCCESS.**
Status reports only due monthly. Advance 3.	**Team takes vacation. Return refreshed, but go back 2.**	*SSMART* **S**TSC **S**tatistical **M**anagement **A**nalysis **R**esponse **T**ool		**Team lead multi-tasked to other projects. Skip 2 turns.**
Holding weekly team meetings. Go ahead 2.	**Can't delete unachievable requirements. Sit out 4 turns.**			**Requirements re-scoped. Roll twice, multiply, and go back that many.**
Daily team meetings that last 10 minutes or less. Go ahead 1.	**Daily status reports required. Go back 6.**		**Earned value metrics ignored, schedule updated unrealistically. Skip 2 turns.**
Daily team meetings that drag on and on. Go back to START.	**Team members added late? Go back 11.**	**SUCCESS** You're done! No good deed goes unpunished. Go back to beginning and start over. Repeat until retirement age.		**No team meetings. Go back 6.**
Team lacks ability to set their own schedule. Sit here 2 turns.	**Running short of money. Time for team party.**	**Team broken up? Square die and go back that many.**		**New reporting metrics required. Go back 5.**
Small team size. Go ahead 4.				**Replace team lead. Go back 2 squares for each time this has occurred.**
Large team size. Skip 1 turn.				**New team members added late. Roll twice and go back that many.**
Forced to use new and untested tools. Post résumé, skip 1 turn.	**New team members added early. Go ahead 1.**	**Weak team leadership. Stay here 2 turns.**	**Team penalized for setting unrealistic schedule. Demotivated team skips 2 turns.**	**Team squabble. Go back 1 for each day team members not speaking.**

S T S C

TEAM
No. 1

TEAM
No. 2

—**David A. Cook, Ph.D.**
Senior Research Scientist and Principal Member of the Technical Staff
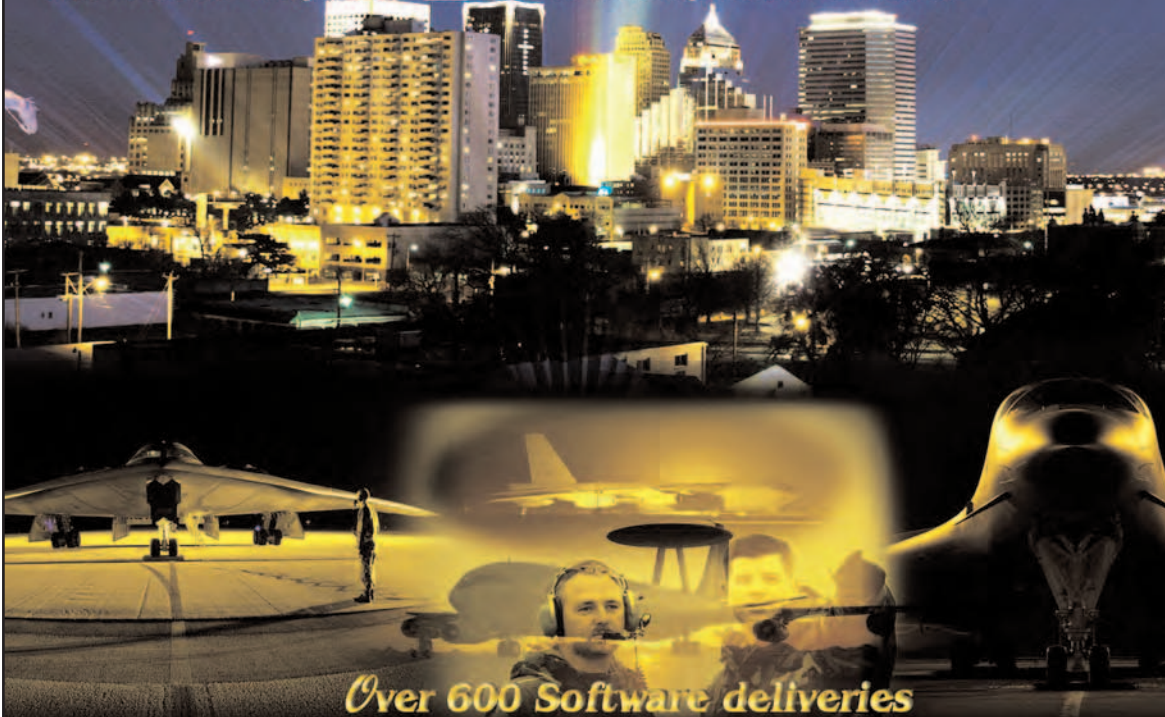*The AEgis Technologies Group, Inc.*
dcook@aegistg.com